

Software per a llegir els llavis (Lip Reading)

Javier Alejandro Camacho Machaca

Resum — Aquest treball presenta el desenvolupament d'un software capaç de llegir els llavis mitjançant el reconeixement d'un rostre. Amb aquest programari es busca facilitar el treball per a aquelles persones amb dificultats visuals o auditives, o en altres casos per a persones que vulguin saber que diu altra persona en una multitud o si està molt lluny. [Explicació de com s'ha aconseguit desenvolupar el software (mètode, eines, investigacions etc...) i dir els seus resultats.]

Paraules clau—Paraules clau del projecte, màxim 2 línies.

Abstract—Versió en anglès del resum.

Index Terms—Versió en anglès de les paraules clau.

1 INTRODUCCIÓ - CONTEXT DEL TREBALL

És ben sabut que les persones amb dificultats o discapacitats visuals o auditives tenen més dificultats per interactuar amb altres persones. Això pot ser degut al fet que la resta de la gent pot no conèixer el llenguatge de signes, o perquè la persona amb la qual volen comunicar-se té dificultats per veure o bé no té capacitat visual. També cal considerar aquelles persones que han perdut la capacitat de comunicar-se o entendre als altres, com ara les persones que pateixen càncer de gola i perden la capacitat de parlar, com ha estat el cas de Val Kilmer (Figura 1). Un altre exemple és el cas de Bruce Willis, que pateix d'afàsia, una malaltia que provoca la pèrdua de la capacitat d'expressar-se o comprendre el llenguatge, com a resultat del dany a les àrees del cervell que controlen el llenguatge. Aquestes situacions poden ser causades per diverses malalties o accidents que canvien la vida de les persones per sempre.

Encara que ja hi ha diverses empreses i centres que han abordat aquest problema, nosaltres volem contribuir a aquesta causa amb el nostre esforç.



Figura 1. Val Kilmer, en el 2015 es va sotmetre a una traqueotomia i avui porta sempre mocador al coll.

Per aquesta raó, hem desenvolupat un software que serà capaç de llegir els llavis mitjançant un reconeixement facial que se centrarà en les expressions dels llavis. Utilitzant xarxes neuronals com ara RNN (Recurrent Neural Network) o LSTM (Long Short-Term Memory), processarem les dades d'un vídeo per convertir-les en una seqüència d'imatges, que serà el nostre veritable input i que serà utilitzat per predir el que la persona objectiu del vídeo està dient. Un cop el vídeo ha estat processat pel nostre model, tenim llibertat per ampliar aquest resultat de diverses maneres. Podríem utilitzar el nostre propi software per a nosaltres mateixos, en cas que no tinguem la capacitat de parlar, però sí de moure els llavis, podem transcriure la predicció del model a text i, gràcies a una API de "text-to-speech", expressar-ho a través d'un altaveu. Donant-li veu a les persones que no ho tenen.

- E-mail de contacte: 1566088@uab.cat
- Menció realitzada: Enginyeria de Computació
- Treball tutoritzat per: Coen Antens (Ciències de la Computació i Intel·ligència Artificial)
- Curs 2023/24

També podríem aplicar aquesta última idea de "text-to-speech", però de manera inversa. És a dir, podríem utilitzar el nostre programari per transcriure el que s'està dient a text mitjançant una API "lips-to-text" (per al nostre cas). Això permetria que una persona es pugui comunicar amb algú que no pot escoltar, fent que pugui llegir el que estem dient. L'opció més viable seria utilitzar les APIs de l'empresa OpenAI, les quals fan ús de la intel·ligència artificial.

Altres casos importants a tenir en compte, més enllà de qualsevol mena de discapacitat, són aquells en què hi ha molta gent i és difícil escoltar el que el teu company vol dir. En aquests casos, el nostre programari no tindria cap problema per llegir els seus llavis. Un altre escenari seria quan vols comunicar-te amb una persona que està molt lluny, i per tant no la pots escoltar ni veure clarament. En aquest cas, si apliquem el nostre programari a una càmera amb una resolució suficient, podríem saber què vol dir aquella persona. També hi ha situacions que van més enllà de la necessitat personal, com ara els comentaristes esportius. Gràcies a la nostra aplicació, tant les persones que no entenen el que diuen per què parlen massa ràpid com les persones sordes, podran gaudir de qualsevol partit.

Per comprendre el problema que planteja la barrera de l'idioma a l'hora de parlar, ens centrarem en les expressions dels llavis per interpretar els seus gestos, ja que aquesta interpretació ens permetrà saber quin so fonètic està produint-se, i la fonètica és similar en els idiomes llatí i germànic. Per aconseguir-ho, entrenarem el nostre model amb un conjunt de dades de vídeos proporcionat per la mateixa BBC (British Broadcasting Corporation) [10]. Aquest conjunt de dades consisteix principalment en frases o paraules en anglès extretes de les transmissions de la BBC News. Va ser necessari sol·licitar permís abans de poder-ne fer ús.

2 OBJECTIUS

Fer un software inicial que classifiqui quina vocal està dient: Per començar a familiaritzar-nos amb les característiques més rellevants de les expressions dels llavis. Farem ús de la llibreria 'Mediapipe' per identificar els rostres de les persones i poder agafar punts de coordenades que seran els punts de referència que conformen la forma dels llavis. Aquesta informació l'utilitzarem per entrenar un model de 'Support Vector Machine' i un altre de 'Random Forest'.

Fer un software que detecti, dins d'un rang, que paraules està dient de manera aïllada: Aprofitant l'experiència i les dades recollides durant el desenvolupament del programa per a la detecció de vocals, iniciarem la creació d'un programa destinat a predir quines paraules està pronunciant una persona en un vídeo. Com aquesta tasca difereix considerablement de la simple detecció de vocals, ens orientarem cap a models de Deep Learning, especialment les RNN (Xarxes Neuronals Recurrents), que processaran els frames del vídeo comentat anteriorment per a predir la paraula que està pronunciant la persona objectiva.

Desenvolupar el software per a la detecció de paraules amb capacitat per classificar un major nombre de paraules de manera més precisa i fluida: Aquesta millora implica enriquir el programa de detecció de paraules aïllades mitjançant un entrenament amb una base de dades més àmplia. Així, serà capaç de predir una gamma més gran de paraules i nosaltres desenvoluparem una forma de què ho faci d'una manera més continuada, permetent la formació de frases com a resultat. Aquest objectiu representa la fita ideal per a aquest treball, ja que abraça de manera integral els nostres objectius en aquest TFG.

3 METODOLOGIA

L'organització i les eines utilitzades en la realització d'un projecte són crucials per aconseguir-ne el màxim rendiment en el temps disponible i per mantenir sempre una referència clara del que s'havia planejat en comparació amb la realitat. En aquest sentit, explicaré les opcions que he decidit utilitzar i les raons darrere d'aquesta elecció.

3.1 Mètode Àgil

La metodologia que hem triat és el Kanban, ja que en tractar-se d'un projecte realitzat per una sola persona, no cal seguir pautes tan estrictes com les que requeriria el mètode Scrum. El sistema Kanban permet gestionar les tasques en tres categories: "Per fer", "En curs" i "Fet". Aquesta metodologia és ideal per gestionar les tasques de manera senzilla i eficaç.

Per tant, farem servir aquest mètode àgil per organitzar les tasques més generals i rellevants d'una manera que sempre sigui clara quant al temps necessari per dur-les a terme i quines subtasques les componen, assegurant-nos que cada tasca es compleixi de la manera més exhaustiva possible.

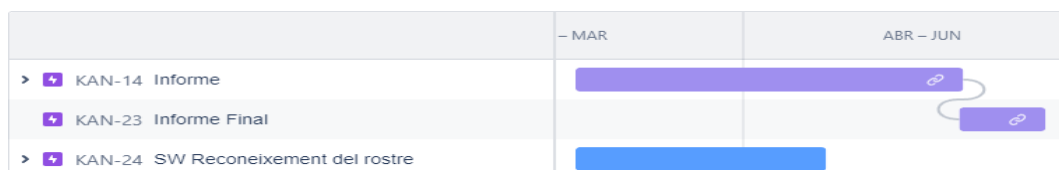


Figura 2. Cronograma exportat de Jira Software

3.2 Eina de seguiment

Per al seguiment i la gestió de les tasques necessàries per desenvolupar aquest software, hem optat per utilitzar Jira Software. Aquesta eina, a més de ser una de les principals en el seu àmbit, compleix tots els requisits necessaris per al nostre projecte. Ens permet disposar d'una interfície on puc aplicar el mètode Kanban, gestionant les meves tasques en subtasques i assignant-les estats com "En curs" o "Finalitzat", al marge del sistema principal de Kanban. A més, em possibilita crear un calendari per establir terminis. Les funcionalitats que hem utilitzat de Jira Software són el "Tauler", que em permet visualitzar les meves tasques com a pòsits i moure'ls pels estats de Kanban (Per fer, en curs i fet). Les "Incidències", on puc veure totes les meves tasques i terminis per entendre-les millor, i on puc marcar-les com a "Finalitzat" o "Per fer". Finalment, el "Cronograma", que em mostra el temps dedicat a cada tasca i el temps restant per a completar-les (Figura 2).

4 ESTAT DE L'ART

Aquest projecte comença amb una investigació i recerca de altres projectes o empreses que hagin realitzat alguna cosa similar a la lectura dels llavis. Els treballs que he pogut trobar que tenen com a objectiu final el Lip Reading són:

LipNet (End-to-End Sentence-level Lipreading) [1]: És un model que assigna una seqüència variable de fotogrames de vídeo a text, fent ús de convolucions espaciotemporals i una xarxa recurrent, entrenat totalment de manera integrada. Segons diuen ells, LipNet és el primer model de lectura de llavis a nivell de frase, integrat de principi a fi, que aprèn simultàniament de característiques visuals espaciotemporals i de models de seqüència.

LipType (A Silent Speech Recognizer Augmented with an Independent Repair Model) [2]: Aquest model es presenta com una versió millorada de LipNet, amb una major precisió i velocitat. Es destaca per un programa de preprocesament d'imatges que s'utilitzen com a entrada, incloent-hi la capacitat de modificar la il·luminació si és necessari, així com corregir els possibles errors en els resultats obtinguts. A més, el model s'implementa utilitzant la llibreria 'Keras' de 'TensorFlow'.

AV-HuBERT [3]: És un model autosupervisat centrat en l'AVS (Audio Visual Speech), és a dir, que té com a entrada tant l'àudio com la imatge. Tot i que l'objectiu és el mateix, la manera de fer-ho és diferent. Tanmateix, això no ha de ser una raó per no tenir-ho en compte a nivell de processament de dades.

Auto-AVSR [4]: És un model de codi obert dissenyat per a l'entrenament end-to-end. Similar al model anterior, aquest se centra en el ASR (Audio Speech Recognition) però també en el VSR (Visual Speech Recognition), sigui de manera individual o combinada. Utilitza la llibreria 'Py-Torch' per a models d'entrenament.

Lipreading using Temporal Convolutional Networks [5]: És un repositori centrat en la lectura dels llavis que utilitza tant l'àudio com les imatges. El que diferencia aquest model de la resta és que ha estat entrenat per una xarxa basada en la convolució 3D i un preprocesament de les imatges del vídeo centrat en els llavis abans de l'entrenament.

Visual Speech Recognition for Multiple Languages [6]: En aquest repositori, com la majoria, s'utilitza de manera individual i combinada l'àudio i el vídeo per al seu entrenament. No obstant això, en aquest cas, es fa servir un model basat en ResNet-18 i un transformador de convolució augmentada (Conformer). Aquest enfocament representa un canvi dràstic en la manera d'afrontar el repte de la lectura dels llavis, ja que, per exemple, fa servir el Conformer en lloc d'una RNN (Recurrent Neural Network).

Tots aquests exemples són bons punts de partida per a iniciar un projecte centrat en el LipReading. Personalment, no seguiré o milloraré alguns d'aquests treballs, però m'han proporcionat idees sobre les eines, llibreries, models o mètodes de processament que puc desenvolupar per fer el meu propi projecte per ser un altre punt de referència en l'àmbit del LipReading.

5 PLANIFICACIÓ

El primer gran repte o objectiu que es va proposar per a aquest projecte va ser desenvolupar un codi capaç de detectar els llavis d'una persona per determinar quina vocal està pronunciant. En primer lloc, vaig utilitzar la llibreria 'Mediapipe', específicament la funció 'face_mesh', per obtenir una malla del rostre. Un cop teníem tots els punts del rostre, ens vam centrar en els punts que formen els llavis per obtenir un mapa de punts que utilitzarem per entrenar un model d'Aprenentatge Automàtic. Posteriorment, vam seguir el mateix procés però fent servir un model de 'Deep Learning' per preparar-nos per a futurs reptes o objectius. El següent pas és desenvolupar un software que em permeti identificar quina paraula individual està dient una persona. Per aconseguir això, hem de fer ús d'altres mètodes i eines més adequats a la nostra problemàtica. Hem considerat que, com una paraula és una seqüència de lletres amb significat, el millor seria utilitzar una xarxa neuronal recurrent, coneguda com a RNN (Recurrent Neural Network). Hem escollit aquesta opció perquè és un tipus d'algoritme de Deep Learning que s'implementa molt adequadament al nostre problema, ja que té com a input dades seqüencials o dades de sèries de temps. L'objectiu final i ideal seria aconseguir identificar les diferents paraules que formen una frase.

Per aconseguir desenvolupar aquest tipus de model, necessitem una base de dades adequada. Hem tingut la sort de poder utilitzar el conjunt de dades 'The Oxford-BBC Lip Reading in the Wild (LRW) Dataset' [10], que conté 1000 expressions de 500 paraules diferents, pronunciades per centenars de parlants diversos de la BBC. Farem servir aquest conjunt de dades per entrenar el nostre model i aconseguir els nostres objectius.

6 DESENVOLUPAMENT

Durant aquest tema, es detallarà el procés de desenvolupament, començant amb la recopilació i la preparació de dades, la implementació de models i algoritmes, i finalment, l'avaluació del sistema desenvolupat. Es discutiran els reptes trobats durant el desenvolupament i es presentaran els resultats obtinguts, destacant les contribucions significatives i les lliçons apreses en el procés.

Per tal de concloure el meu primer objectiu d'elaborar un programari que classifiqui les vocals, he utilitzat la funció 'FaceMesh()' de la llibreria 'Mediapipe'. Aquesta funció, en primer lloc, detecta els rostres de les persones i després retorna una malla de punts del rostre expressada en coordenades i profunditat $[x, y, z]$. Ens centrarem únicament en els 80 punts que formen els llavis per tal d'obtenir el gest en si mateix (Figura 3).



Figura 3. Foto feta per mi mateix i processada per la funció 'FaceMesh()'

Per poder generar una entrada per entrenar el model que implementi aquesta funció, he hagut de crear el meu propi conjunt de dades. Per fer-ho, he gravat diversos vídeos amb diferents distàncies i fons mentre vocalitzava les vocals. Utilitzant l'eina de reproducció de vídeos 'VLC media player', he convertit els fotogrames d'aquests vídeos en imatges i els he classificat segons la vocal que s'està pronunciant. Aquest conjunt de dades és on he implementat la funció i on hem extret els 80 punts que defineixen la forma dels llavis en cada imatge.

Per obtenir un conjunt de dades adequat per a l'entrenament, afegiré un punt addicional que estarà situat al centre de la boca. Per aconseguir-ho, simplement calcularé la mitjana dels valors dels 80 punts existents, obtenint així el

punt 81 també en format $[x, y, z]$. Per prescindir del format de llibreria dels punts i, així, poder dur a terme l'entrenament del model, substituiré les coordenades dels punts per la distància respecte al punt 81, que representa el centre de la boca. D'aquesta manera, en finalitzar aquest procés, disposarem només de 80 distàncies per a cada imatge que analitzem. A partir d'aquí, només queda entrenar el model.

Després d'haver dut a terme aquesta primera tasca, vaig adonar-me que les prediccions del meu model variaven segons la distància a la qual em trobava de la càmera. Quan estava lluny, només podia detectar les vocals 'O' i 'U', mentre que quan estava a prop, era impossible detectar aquestes últimes. Per resoldre aquest problema, vaig desenvolupar dues opcions. La primera consistia a ajustar les coordenades extrems de la funció 'FaceMesh()' en funció d'un marc que envoltava el rostre de la persona a la imatge, en lloc de la mida de la mateixa imatge. La segona solució, que va donar millors resultats, va ser tenir en compte la distància interpupilar, és a dir, la distància mitjana entre les pupil·les de les persones. Per fer-ho, vaig modificar el meu codi per extreure dos punts addicionals de la malla facial generada per la funció 'FaceMesh()', corresponents a la meitat de cada ull. Vaig calcular la distància entre aquests dos punts dels ulls de la mateixa manera que ho feia amb els 80 punts dels llavis al centre de la boca. A continuació, vaig dividir aquesta distància per la mida real de la distància interpupilar, obtenint una escala de mida real per píxel. Amb aquesta escala, vaig ajustar les 80 distàncies que representaven la distància al centre de la boca. D'aquesta manera, independentment de la proximitat o la distància a la càmera, les distàncies es van ajustant adequadament.

Els models d'aprenentatge que vaig utilitzar per a entrenar en aquest primer objectiu van ser el SVM (Support Vector Machine) i el Random Forest. Per avaluar la qualitat del seu entrenament, vaig fer servir el 'GridSearchCV()' i la validació creuada ('Cross Validation') pel cas del SVM (Figura 5), mentre que pel cas del Random Forest vaig analitzar la seva precisió ('accuracy') en la predicció (Figura 6).

Per visualitzar el resultat d'aquesta primera tasca, farem servir la llibreria 'Gradio', que mostrarem a la Figura 4. Gràcies a Gradio, podem crear fàcilment una interfície web per mostrar els resultats d'un model d'aprenentatge, aquestes característiques el converteix en una eina perfecta per a demostracions de les capacitats del model.

Clasificador de Vocales

Carga una imagen y el modelo predecirá su vocal.

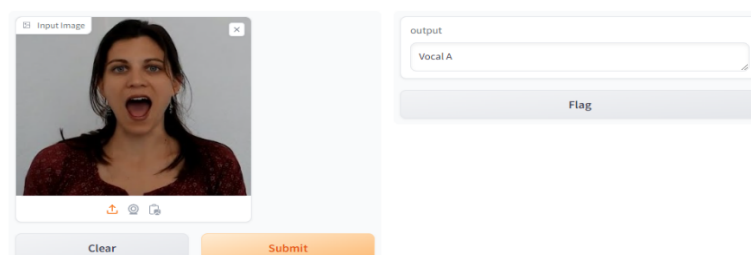


Figura 4. Imatge extreta del resultat de Gradio amb el model de reconeixement de vocals exportat utilitzant el model de SVM (Support Vector Machine). Selecciona una imatge i obtindràs com a sortida quina vocal està pronunciant.

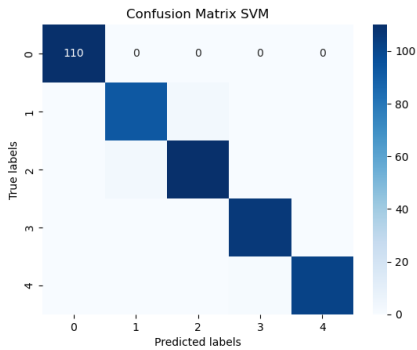


Figura 5. Matriu de confusió del model SVM, entrenat amb els paràmetres extreïdes del 'GridSearchCV()'

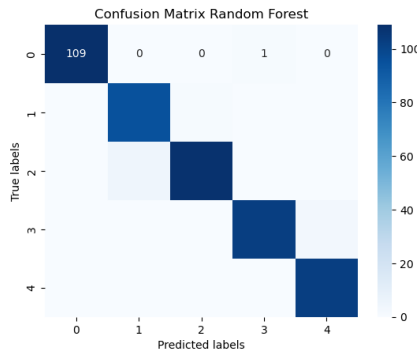


Figura 6. Matriu de confusió del model Random Forest

Per fer-ho més interessant i preparar-me per a l'objectiu final, també utilitzaré aquest mateix procés de creació del conjunt de dades com a entrada per a un model de xarxes neuronals. En aquest cas, faré servir les dades obtingudes mitjançant la metodologia que escala les distàncies en funció de la distància entre els ulls de les persones. Com que l'objectiu és predir bàsicament 5 grups clarament diferenciats, faré servir la biblioteca 'Keras', que proporciona blocs modulars per al desenvolupament de models complexos d'aprenentatge profund. Utilitzaré aquesta llibreria per distingir entre diferents grups de dades que comparteixen similituds, com per exemple predir el tipus de flors en una imatge. Després d'entrenar el model indicant-li que només hi ha 5 grups de dades diferents, vaig observar que les prediccions eren sempre les mateixes, és a dir, que el model només donava com a resposta una de les cinc vocals. Per avaluar la qualitat del model, vaig generar una Matriu de Confusió, que indica si les prediccions són correctes i en quina mesura fallen les prediccions errònies (Figura 7).

A primera vista, la matriu de confusió sembla prometedora pels seus resultats, però en provar-ho en casos reals, no proporcionava una resposta correcta en la majoria dels casos. Després d'intentar trobar la raó d'aquest problema, vaig reflexionar sobre la meua situació, i això em va fer adonar de dues coses. En primer lloc, per enfrontar-me al repte de predir paraules, hauria de replantejar el problema d'una manera diferent de com ho havia fet per predir vocals. En segon lloc, vaig veure la necessitat d'adquirir un conjunt de dades més gran i més adequat per evitar tants

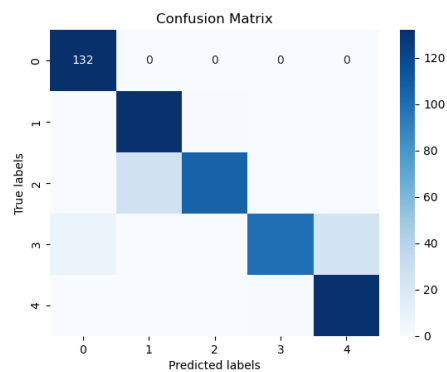


Figura 7. Matriu de confusió del model entrenat amb la llibreria 'Keras'

problemes durant l'entrenament del model. Per aquestes raons, vaig trobar les xarxes neuronals recurrents i el 'The Oxford-BBC Lip Reading in the Wild (LRW) Dataset' [10], que ja havia anomenat en la meua planificació. Vaig optar per aquestes opcions, ja que predir paraules de la mateixa manera que ho havia fet amb les vocals no seria ni adequat ni eficient.

Per començar amb l'objectiu de predir paraules individuals, primer he d'extreure les dades proporcionades per l'organització de la BBC. El dataset està compost per 500 carpetes, cadascuna corresponent a una paraula en anglès, i cada carpeta conté tres subcarpetes: una per a proves, una altra per a l'entrenament i l'última per a la validació. Dins d'aquestes subcarpetes hi ha vídeos, i cada vídeo té un arxiu '.txt' associat amb el mateix nom que el vídeo '.mp4'. Aquest arxiu '.txt' conté cinc línies amb informació rellevant, però l'única línia que ens interessa és l'última, que indica la durada de la paraula pronunciada per la persona del vídeo. Aquesta informació és crucial, ja que tots els vídeos del dataset tenen la mateixa durada exacta, 1,16 segons, o 29 frames. Gràcies a aquesta informació, podem calcular l'inici i el final exactes del fragment que volem extreure dins d'aquests 1,16 segons, pel fet que els vídeos contenen altres continguts a més de la paraula objectiu.

Començant a dur a terme aquesta tasca, va sorgir un problema, vaig adonar-me que els fragments de paraula que volia extreure dels vídeos no sempre estaven exactament al mig, com assegurava la pàgina web del dataset. Aquesta situació complicava les coses, ja que significava que les paraules podien trobar-se més al principi o al final dels vídeos, situacions que no es podien detectar a simple vista.

7 CONCLUSIÓ

.....
.....
.....
.....

AGRAÏMENTS

.....
.....

.....

BIBLIOGRAFIA

- [1] Yannis Assael, Brendan Shillingford, Shimon Whiteson and Nando de Freitas (2016, Novembre) 'LipNet' [Online] URL: (<https://arxiv.org/abs/1611.01599>)
- [2] Inclusive Interaction Lab (2021, Gener 12) 'LipType' [Online] URL: (<https://github.com/theilab/LipType>)
- [3] Meta Research (2022, Gener 6) 'AV-HuBERT' [Online] URL: (https://github.com/facebookresearch/av_hubert)
- [4] Pingcuan Ma (2023, Juny 16) 'Auto-AVSR: Lip-Reading Sentences Project' [Online] URL: (https://github.com/mpc001/auto_avsr)
- [5] Pingcuan Ma (2020, Juny 25) 'Lipreading using Temporal Convolutional Networks' [Online] URL: (https://github.com/mpc001/Lipreading_using_Temporal_Convolutional_Networks)
- [6] Pingcuan Ma (2022, Març 1) 'Visual Speech Recognition for Multiple Languages' [Online] URL: (https://github.com/mpc001/Visual_Speech_Recognition_for_Multiple_Languages)
- [7] OMES (2021, Maig 26) 'Malla Facial (MediaPipe Face Mesh) | Python - MediaPipe - OpenCV' [Online] URL: (<https://www.youtube.com/watch?v=TCUi-pOXuCBQ&list=PLcz5q4ASTYQGKfBzjFBX5nVPkv-TUYn2Ro&index=3>)
- [8] Nachiketa Hebbar (2021, Abril 20) 'Image Classification Web App in Python| Keras +Gradio' [Online] URL: (https://www.youtube.com/watch?v=aZ4wV4V_p9E)
- [9] Spanish Pronunciation Academy (2017, Maig 24) 'Los sonidos del español: Vocal A' [Online] URL: (<https://www.youtube.com/watch?v=E7PKLMBAUtk>)
- [10] J. S. Chung, A. Zisserman (2016) 'The Oxford-BBC Lip Reading in the Wild (LRW) Dataset' [Online] URL: (https://www.robots.ox.ac.uk/~vgg/data/lip_reading/lrw1.html)

APÈNDIX

A1. SECCIÓ D'APÈNDIX

.....

A2. SECCIÓ D'APÈNDIX

.....

