INDEXACIÓ DE CONTINGUTS TELEVISIUS: ANÀLISI DEL SO

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per Sergi Espinar Fernández
i dirigit per Jordi Vitrià Marca
Bellaterra, 13 de juny del 2007.

Universitat Autònoma de Barcelona

Escola Tècnica Superior d'Enginyeria

El sotasignat, Jordi Vitrià Marca

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en Sergi Espinar Fernández

I per tal que consti firma la present.

Signat: .....

Bellaterra, 13 de juny del 2007

ÍNDEX

Avís: El punts 1 i 6.4 de la memòria ha estat realitzat conjuntament amb el meu company Jordi Hernández, degut que tots dos estem realitzant el mateix projecte, però des de dos punts de vista diferents.

INTRODUCCIÓ

En aquest apartat s'exposaran les meves motivacions a l'hora d'escollir aquesta temàtica pel projecte de final de carrera, així com els objectius que varem escollir, ja que és un tema prou complex i pot ésser enfocat de diverses maneres.

També s'explicarà breument la organització d'aquest document.

1 Introducció

1.1 Motivacions

Una de les principals motivacions que ens ha portat a fer aquest projecte és que ens sembla sorprenent veure com un ordinador pot arribar a aprendre i a fer coses que fa uns anys semblaven de ciència ficció. D'altra banda la televisió ha estat present en les nostres vides des de sempre i creiem que ara esta a punt d'experimentar grans canvis respecte el que estem acostumats a veure.

La visió per computador, juntament amb la intel·ligència artificial, és una de les àrees que més ens atreuen en el camp de la informàtica ja que la visió és un dels fenòmens naturals més interessants i complexos, però a la vegada necessari. Moltes vegades s'ha imitat el comportament de la natura amb gran èxit i encert. N'és un clar exemple la visió dels ratpenats, del quals s'ha copiat el seu funcionament en els sonars dels submarins.

Creiem que relacionant la visió per computador, la intel·ligència artificial i la televisió, podem arribar a assolir grans objectius.

Amb aquest nou futur imminent que li espera a la televisió, l'ús d'aquestes branques de la informàtica ajudaran a crear la nova revolució de la informació que estem a punt de viure.

1.2 Indexació de continguts

1.2.1 Què és?

La indexació de continguts consisteix en l'etiquetatge de material (ja sigui material audiovisual, pàgines web o qualsevol altre contingut) de manera que la cerca del material que ens interessa es faci de manera ràpida i eficient a partir d'aquestes etiquetes prèviament creades.

És un concepte molt senzill, però a la vegada és una eina clau en la ordenació de la informació. Si disposem de grans quantitats de dades i d'informació es fa totalment imprescindible disposar d'algun mètode per a la recuperació de la informació que desitgem.

Aquestes etiquetes que associem a la informació sovint són anomenades metadades (o metadata) . Metadata no és res més que dades sobre les dades. Pot semblar una definició

buida i sense sentit, però realment és informació (dades) sobre un recurs concret (dades).

1.2.2 Com es realitza la indexació?

La indexació de continguts es pot realitzar de diverses formes. A continuació expliquem com es pot realitzar aquesta feina:

- Indexació manual: És el mètode més simple i que requereix més temps per part de les persones encarregades de gestionar les etiquetes. Consisteix en trobar aquestes característiques importants que defineixen el contingut i etiquetar-ho manualment mitjançant una base de dades.
- Indexació social: Aquest mètode és també manual, però es diferencia de l'anterior en el fet que els usuaris de la base de dades tenen la possibilitat d'etiquetar el material en el moment en que el visualitzin. L'etiquetatge no és realitzat per un grup de persones tancat. Aquest mètode està guanyant popularitat gràcies a la forta expansió que ha sofert Internet en aquest últims anys. L'exemple més clarificador d'aquesta tècnica d'etiquetatge el trobem en el portal de vídeos YouTube [24].
- Indexació automàtica: Aquí és on realment entren en joc les diferents eines que ens proporciona la intel·ligència artificial i la visió per computador. Aquest mètode consisteix en la indexació automàtica dels continguts mentre aquests son capturats, o bé amb un processament de les dades posterior, mitjançant tècniques informàtiques que siguin capaces d'extreure les característiques diferenciadores del contingut.

1.2.3 Tipus d'indexacions

Segons el nivell en que analitzem les dades podem dividir la indexació en els següents grups:

- Indexació d'alt nivell: En aquest tipus d'indexació el que s'emmagatzema com a metadades és el contingut d'alt nivell del material. Aquest alt nivell normalment consisteix en l'acció, el temps i l'espai i altres dades referents al significat del material.

- Indexació de baix nivell: Aquest altre tipus d'indexació no treballa en el significat del material sinó en les seves característiques bàsiques. En el cas que estiguem analitzant material audiovisual, aquestes dades poden consistir en el color mig de la imatge, el to i sonoritat mitjana del so, o bé els nivells de contrast i saturació.
- Indexació en un domini específic: Aquesta tècnica fa servir les característiques d'alt nivell per a obtenir les dades de baix nivell. Aquest tipus de tècniques només resulten efectives en un domini d'aplicació molt específic, el qual és una limitació prou important.

1.2.4 Origen i usos de la indexació

Actualment hi ha un gran nombre d'empreses que treballen amb grans quantitats de dades, com ara les dades audiovisuals. Sovint tenen un gran repositori d'informació on guarden l'històric dels diferents treballs realitzats, així com material necessari per la realització dels productes. En les empreses el temps és or, i és necessària la cerca ràpida del material que es desitja i la recuperació d'aquest en el mínim temps possible. No és viable navegar per les carpetes del repositori fins a trobar el material que es necessita. Per tant, cal un mètode que solucioni aquestes mancances en l'àrea de recerca de grans quantitats d'informació. La solució adoptada per la majoria consisteix en relacionar els continguts amb metadades.

Així doncs, si una agencia de publicitat desitja recuperar un anunci realitzat fa uns quants anys en el que apareixia un nen que menjava un gelat, faran servir un buscador que a partir de les paraules clau "nen" i "gelat" trobarà l'anunci en un temps acceptable. En canvi, imaginem que haguéssim de mirar els anuncis realitzats en els últims anys per a localitzar-lo, seria totalment ineficient i poc productiu.

En l'actualitat, la indexació de continguts no s'ha limitat únicament a l'ús empresarial. En el marc d'Internet, és ben coneguda la pàgina Youtube [24]. Aquesta pàgina permet la recerca i visualització de vídeos a partir de certes paraules clau que introduïm en el buscador. Aquestes paraules clau consisteixen en metadades sobre el vídeo a visualitzar. Cada usuari que decideix posar un vídeo a la xarxa definirà les paraules clau sobre l'arxiu i el pujarà a la web per a ser accessible per a tothom a qui li interessi.

1.2.5 El futur de la indexació

Amb la forta demanda dels continguts multimèdia, la indexació està creixent i guanyant gran popularitat. Gran part d'aquesta popularitat es deguda a l'ús d'Internet.

Inicialment, la indexació de pàgines web per a ser localitzades pels buscadors, la feien els mateixos creadors de la pàgina, això va ser el que va posar els ciments a tot el procés que vindria després. És clar que quan va començar tot, la tecnologia de l'època no permetia compartir continguts multimèdia degut a la lentitud de les xarxes de comunicació i que els formats de compressió d'arxius multimèdia es trobaven a les beceroles.

Els temps han canviat, i actualment la compartició d'arxius multimèdia a través de la xarxa gaudeix de bona salut. És evident que aquest ha estat un punt d'inflexió que ha contribuït notablement al creixement de la indexació i la gestió de continguts.

Creiem que el futur a curt termini que li espera a la indexació és realment impactant. Amb la entrada de la nova televisió digital i interactiva, on disposarem d'una quantitat de canals increïblement gran, serà necessari un mètode per a seleccionar què volem visualitzar segons el que ens vingui de gust en aquell moment.

Amb una amplia oferta de canals com prometen que tindrem, al voltant d'uns cinc mil, practicar el Zapping serà poc productiu, ja que quan haguem visualitzat l'últim d'aquests canals segurament ja haurà canviat la programació del primer que haurem vist. És a dir, la gestió de continguts serà més una necessitat que no pas una possibilitat.

1.3 Televisió

1.3.1 On ens trobem en la televisió

Avui en dia, encara predomina la televisió analògica, on tan sols ens arriba una senyal de vídeo i veu, més unes dades de teletext que s'aprofiten a enviar en els moments de rastreig de la imatge. A més, no hi ha cap forma fàcil i ràpidament accessible per poder enviar dades en la direcció oposada, es a dir, de casa nostra a la cadena de televisió. Això és un gran inconvenient tenint en compte com està el món actualment, ja que cada vegada més, es vol una interacció entre l'espectador i la cadena. La forma tradicional de fer aquesta comunicació és utilitzant el telèfon, i actualment també els missatges SMS o

els correus electrònics. Tot i així, encara hi ha un gran desfasament comparat amb la gran revolució en la forma de comunicació humana d'aquest segle, Internet.

És per això que ha sorgit la TDT, televisió digital terrestre, la qual intenta poder cobrir aquestes mancances. La TDT permetrà una millora considerable de la imatge i el so, utilitzant molt més amplada de banda per poder transmetre el seguit de dades, i a més, també permetrà transferir altres dades digitals per poder fer una interacció molt més còmoda amb l'usuari i la cadena.

Tot això es pot aconseguir fent un aprofitament de l'amplada de banda gracies a transmetre la informació digitalment, la qual cosa ens permet poder comprimir totes les dades a ser enviades. Gràcies a la computació d'avui en dia no suposa cap pèrdua considerable de temps.

Algunes d'aquests serveis interactius que la TDT ens permet, combinant-ho amb l'estàndard MHP, són les Guies de programació electròniques (EPG), anuncis interactius i serveis d'informació com ara les últimes noticies, el temps de la teva zona, informes del tràfic, etc.

S'ha de tenir en compte que actualment la televisió està perdent terreny respecte a Internet, ja que ja hi ha molta gent que prefereix passar el seu temps lliure navegant per la web i mirant els continguts que ell tria en cada moment que no pas estar davant de la televisió per veure que emeten. A més, amb la capacitat d'amplada de banda disponible actualment, ja és possible la descàrrega de programes o series per Internet i veure-les justament quan vols, evitant dependre de l'horari de la televisió. Aquest serà un fet que haurà de canviar en els pròxims anys si la televisió vol seguir sent dels primers en el negoci de l'entreteniment.

1.3.2 Cap a on avança la televisió

És sempre incert el que ens depara el futur, però es poden intuir certs aspectes que segur que predominaran en la futura televisió. Una de les principals novetats serà la televisió sota demanda, en la qual es podrà tirar que veure en cada moment, no com ara que hi ha una programació lineal i has d'estar a l'hora que comença el programa per poder-ho veure, sinó que hi haurà com una mena de carta televisiva on podràs tirar el que vols veure i quan ho vols veure. De fet, ara ha sortit un programa que fa exactament això, es

diu Joost [25]. Encara està a la fase inicial, però els creadors afirmen que serà una revolució en la forma de veure la televisió.

Una altra novetat completament relacionada amb la que acabem de comentar, serà com triar el que vols veure a cada moment. Aquí és on entra clarament el nostre projecte, ja que hi haurà d'haver mètodes de cerca basats en contingut molt més específics que no els que hi ha avui en dia, ja que hi haurà una gran quantitat de possibles opcions a l'hora de triar el que vols veure.

Altres novetats que revolucionaran el món de la televisió, seran una interacció molt més lleugera entre la cadena i l'espectador, permetent la comunicació a temps real, com per exemple programes de concursos interactius amb tots els espectadors, on des de casa amb el telecomandament a distància podran marcar la resposta correcta i fins i tot guanyar premis. També hi podrà haver comunicació entre espectadors, com fòrums de debat directament incrustats en el mateix programa.

També hi haurà una comunicació totalment directa entre la televisió i Internet, sent possible tenir un navegador web just al costat del programa que estem mirant, tenir un programa de missatgeria instantània per poder a la vegada parlar amb els teus amics o que s'obrís una nova finestra quan arribés un correu electrònic nou, etc.

Els anuncis publicitaris de televisió, tal com els concebem actualment, per força hauran de canviar. Com que són la principal font d'ingrés de les cadenes televisives, farà falta buscar nous mètodes per poder posar la publicitat. Un mètode que s'està proposant actualment, no per solucionar aquest problema, però si per crear una forma d'anunciar més eficient i potser menys pesada, és utilitzar el mètode que fa servir Google amb el seu sistema de publicitat en pàgines web, Google Ads. Aquest mètode consisteix en posar anuncis relacionats amb el contingut que estàs visualitzant.

Per una banda, a l'afegir interacció amb els programes, es podran crear anuncis molt més dinàmics, i més orientats a l'espectador en concret. Es podran emetre anuncis diferents per cada usuari, fent que aquests corresponguin a les aficions i interessos de l'usuari concret. Això es podria realitzar mantenint un historial dels programes preferits per poder crear un perfil d'usuari i mostrar anuncis que poguessin ser del teu interès. Inclús es podran recollir dades de les reaccions dels usuaris davant d'un anunci en concret (si decideix interactuar, si simplement l'observa, si canvia de canal...).

Per altra banda, la gent intentarà evitar els anuncis tant com sigui possible. Existeixen sistemes de detecció d'anuncis, com ara els plantejats a [4]. Això farà que els usuaris evitin els anuncis, suprimint-los de les seves gravacions televisives. Les empreses de publicitat hauran d'innovar i plantejar nous mètodes de publicitat, ja que sinó ningú veurà el seu treball.

Amb la nova televisió digital, on es podran descarregar els programes a més velocitat que no pas a temps real, la eliminació dels anuncis i la publicitat sense haver d'esperar a disposar d'una gravació (en cinta VHS, DVD o disc dur) serà una realitat.

Segurament és pot pensar la televisió del futur, com una barreja entre Internet i la televisió convencional, la qual cosa ens porta directament a pensar que hi haurà noves cadenes de televisió, nous programes, que ara són del tot impossibles. La televisió d'avui en dia està pensada per agradar a gent geogràficament propera, i per nacions. Amb Internet la cosa canvia, i hi haurà una televisió per a tot el mon, la qual cosa permetrà crear programes que aquí només serien d'interès d'uns quants, però en tot el món, seran una gran quantitat a tenir en compte.

1.3.3 Gestió de continguts

Hem vist doncs, que un punt clau d'aquesta nova televisió serà el de poder triar els programes del nostre gust a cada moment. Per tant, la indexació de contingut serà un dels temes més importants a tenir en compte per a poder cercar còmoda i fàcilment els programes que desitgem. No cal dir que també serà de vital importància establir quines poden ser les característiques a indexar, ja que en un entorn on hi poden haver més de 5.000 canals amb diferents programes a triar a qualsevol hora, amb quatre característiques no en tenim ni per començar, ja que només podríem tenir un nombre petit de tipus de programes diferents i no s'ajustaria massa a les diferents necessitats de cada persona.

És per això que cal buscar noves característiques per a indexar, però ens porta a un nou problema, com indexar tota aquesta gran quantitat de dades. La solució recau en la informàtica, en utilitzar tècniques de visió per computador i intel·ligència artificial per poder fer una indexació automàtica eficient i si pot ser, ràpida.

1.3.4 Convergència de tecnologies

Com ja hem comentat anteriorment, el que segurament el futur ens prepara, és una convergència entre totes les tecnologies, amb Internet com a l'element intermediari. És un gran pas que encara trigarà uns anys a ser del tot quotidià, però el més probable és que tingui una gran acceptació per part de tothom. Així doncs, se'ns obre un ventall enorme de possibilitats, ja siguin d'interacció amb la televisió com de tasques totalment automàtiques a la hora de buscar programes. Serà possible, per exemple, poder utilitzar el telèfon mòbil per comunicar-se amb l'orinador de casa, perquè aquest a la vegada es connecti amb el televisor i comenci a buscar per tu el programa que més de gust et pot venir. Així només arribar ja ho tindràs preparat. Aquest fet pot ser del tot viable si tenim en compte tot el que hem comentat en aquestes pàgines. Un perfil d'usuari amb els gustos, una televisió digital on es poden veure tots els programes quan vols, més un sistema d'indexació i cerca de continguts ens porta directament a aquest futur no tan llunyà.

També podríem passar a ser esporàdicament els protagonistes d'un concurs a través de la webcam o amb una simple resposta del comandament a distància.

Gràcies a aquestes noves possibilitats, sorgiran noves aplicacions per fer cada vegada més interactiva la televisió, utilitzant les diferents tecnologies ja conegudes i les que estan per venir. Això ens portarà a un sistema d'entreteniment completament nou i ple de noves experiències.

Clars exemples d'aquesta convergència de tecnologies els podem trobar a la cadena de televisió americana NBC, que permet descarregar-se els capítols de totes les sèries que emeten per a ser visualitzats amb un iPod.

És un gran encert per part d'aquesta cadena de televisió. És evident que si ells no posen disponibles aquestes sèries per Internet, algú altre s'encarregarà de fer-ho.

Per altra banda, trobem l'exemple de Joost [25], esmentat anteriorment, un projecte dels creadors de Kazaa i Skype. Permet visualitzar continguts audiovisuals en Streaming a través d'Internet. Aquests continguts són compartits per els diferents usuaris de la aplicació, com si es tractés d'un programa d'intercanvi d'arxius P2P.

1.4 Treball

1.4.1 Explicació dels objectius inicials

L'objectiu principal d'aquest projecte és poder demostrar que hi ha certes característiques en els programes de televisió (ja siguin pel·lícules, sèries, programes, telenotícies...) que ens han de permetre poder classificar aquests programes per altres categories apart de les característiques típiques de gènere, actors principals, idioma, etc. sinó que se'n poden extreure d'altres més relacionades amb sensacions i sentiments. Tenint en compte que s'acosta la nova televisió digital, on buscar programes del nostre gust serà del tot habitual, poder basar-se en més característiques que no pas les típiques serà molt interessant.

A més, hi ha molts programes semblants de contingut, uns que triomfen completament i d'altres en canvi, que queden en l'oblit. Això és degut segurament, a característiques imperceptibles per nosaltres mateixos, però que si féssim una recerca exhaustiva utilitzant intel·ligència artificial, veuríem que hi són, i que hi tenen molt a veure. Si sabéssim exactament quines són les variables del subconscient que fan que ens agradi més un programa que l'altre, i que fan que un programa tingui més audiència que un

altre, podríem arribar a entendre una mica més com funcionen els gustos dels humans i tirar endavant una línia de recerca completament nova pel que fa a aquest aspecte.

Nosaltres ens hem basat en els telenotícies, ja que és un programa comú en totes les cadenes amb diferents formats, convertint-se en un clar exemple per poder diferenciar aquestes trets esmentats anteriorment. En Jordi Hernández ha treballat la part d'anàlisi de la imatge, mentre que el Sergi Espinar s'ha centrat en l'anàlisi del so. La fusió d'aquests dos treballs farà possible la creació d'un sistema complet d'anàlisi i extracció de característiques.

Tot i que aquest és un gran objectiu, nosaltres limitarem el nostre treball a una primera fase, que serà crear un sistema ràpid per poder extreure tot un seguit de descriptors i demostrar que és possible diferenciar i classificar els telenotícies a partir d'unes certes característiques que creiem que poden influir en les esmentades sensacions.

D'altra banda també volem demostrar que les diferències s'accentuen en cadenes de televisió amb més i menys pressupost, i veure que és el que ens fa veure de seguida de que es tracta d'una d'aquestes.

1.4.2 Estructuració del treball

Aquesta memòria està estructurada en quatre grans blocs:

El primer bloc és aquesta introducció, on expliquem les motivacions que ens portat a la creació d'aquest projecte, els objectius que hem volgut assolir i una breu descripció de l'organització d'aquest document.

Seguidament trobem l'apartat de mètodes, on expliquem els coneixements teòrics necessaris que he emprat per fer el projecte, incloent el perquè i com els hem utilitzat.

A continuació hi ha l'apartat de la implementació on expliquem detalladament el procediment que hem seguit per a la realització del projecte, els diferents programes que hem fet i la relació entre ells, així com els resultats obtinguts en els nostres experiments.

Finalment tenim l'apartat de les conclusions, on resumim el que s'ha aconseguit i el que es podria millorar del treball d'aquests darrers mesos, així com les principals aportacions del nostre treball i les diverses línies de continuació d'aquest projecte.

Hem de recalcar que aquest projecte consta de dos parts ben diferenciades però que estan fortament relacionades, a més hem treballat conjuntament a l'hora de planificar el projecte i sobretot en la part final del classificador, la qual cosa ens ha dut a fer una memòria amb algunes parts compartides i d'altres de molt semblants, com ara aquesta introducció i l'apartat dels resultats conjunts.

MÈTODES

Aquest apartat explica els fonaments teòrics en que es basa el projecte, així com les diferents tècniques actuals que afecten el tractament del so mitjançant tècniques informàtiques.

2 Mètodes

2.1 Music Retrieval

Una de les aplicacions obvies que li podem donar a l'àudio extret de la televisió és el reconeixement d'aquests sons i una posterior classificació. Poden existir moltes maneres per a reconèixer sons, però sens dubte, la més innovadora és la desenvolupada per Yan Ke, Derek Hoiem i Raul Sukthankar [1]. Aquest mètode té la particularitat que no treballa directament amb el so generat, sinó amb la imatge que genera la seva resposta en freqüència, és a dir, l'espectrograma. Es passa d'una interpretació del problema en una dimensió, a la representació en dues dimensions. L'origen d'aquesta tècnica es troba en la representació en imatges que realitzen del so els investigadors d'aquest camp. Aquesta representació és utilitzada tradicionalment per a visualització humana, però utilitzant certes tècniques de visió per computador es pot realitzar una identificació del so molt robusta i fidedigna. La realització de l'espectrograma està explicada en un document annex a aquesta memòria.

2.1.1 Funcionament

Aquest és un sistema molt resistent al soroll, ja que permet, a partir d'una gravació de mala qualitat, com ara un arxiu d'àudio gravat amb un micròfon, reconèixer el so amb una fiabilitat del 90% amb les pitjors condicions.

Un altre dels avantatges d'aquest mètode és que no requereix grans quantitats de dades per a identificar el so. En una gravació de tres segons hi ha informació suficient com per a reconèixer el so amb una elevada fiabilitat.

L'altra particularitat que fa especialment interessant aquest mètode és la velocitat en que treballa, així com la ràpida resposta que ofereix el sistema. La tècnica usada per a convertir la senyal del domini temporal en un espectrograma i realitzar una comparació fent servir tècniques de correlació és lenta i inexacta. En canvi, el que proposa aquest mètode és utilitzar una petita quantitat de filtres que tenen una resposta robusta davant del soroll, i a la vegada preserven la informació necessària per a fer la distinció entre els diferents sons. Aquests filtres no son definits manualment, sinó que es defineix un conjunt gran de filtres candidats, i amb tècniques d'aprenentatge computacional s'identifica un petit subconjunt que funcionen molt bé plegats.

Per a trobar la família de filtres adequada per a la resolució del nostre problema, resulta prou útil examinar les característiques dels espectrogrames que els diferencien entre ells. Aquestes son les següents:

- Diferencies de potència en bandes de freqüència veïnes en un instant determinat.
- Diferències de potència a través del temps a l'interior d'una banda de freqüència particular.
- Desplaçaments de la freqüència dominant a través del temps.
- Pics de potència a les freqüències en un instant determinat.
- Pics de potència a través del temps a l'interior d'una banda de freqüència particular.

La família de filtres proposada ha de ser capaç de capturar aquestes característiques mentre estem treballant a través de diferents bandes de freqüència amb amplades de banda diferents i amb diferents extensions de temps. Els filtres amb una gran amplada de banda i temps ofereixen més robustesa a certes distorsions, però els filtres amb una petita amplada de banda i temps poden capturar informació discriminant que no poden adquirir els altres filtres.

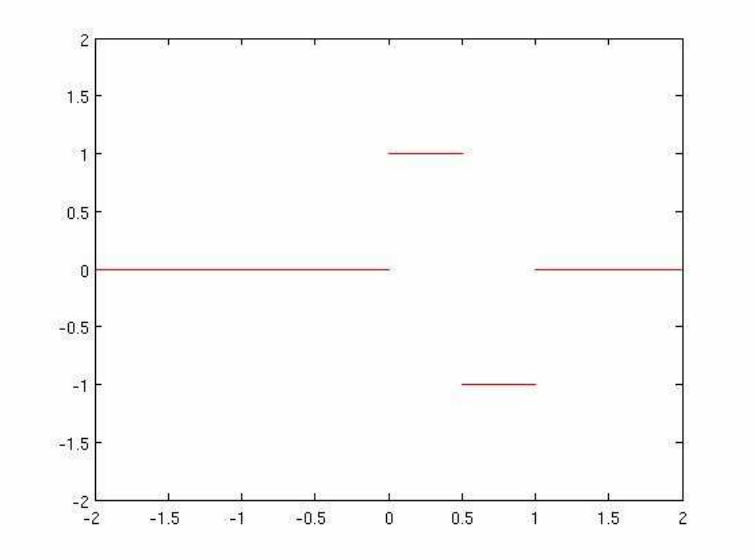
Si veiem l'espectrograma com una imatge normal i corrent en escala de grisos, veiem que els filtres Wavelet de la classe de Haar compleixen els requeriments que necessitem.

Una sèrie de Wavelets és una representació d'una funció integrable quadràtica (la integral del quadrat del seu valor absolut durant un interval concret és finita). Els Wavelets de Haar van ser els primers Wavelets coneguts. És el Wavelet més simple que existeix. Té l'inconvenient de que no és una funció contínua, i no és integrable.

És una funció graó de la forma següent:

$$f(x)\begin{cases} 1 & 0 \le x < 1/2 \\ -1 & 1/2 \le x < 1 \\ 0 & \text{en qualsevol altre cas} \end{cases}$$

La representació gràfica és la següent:



Representació gràfica del Wavelet de Haar.

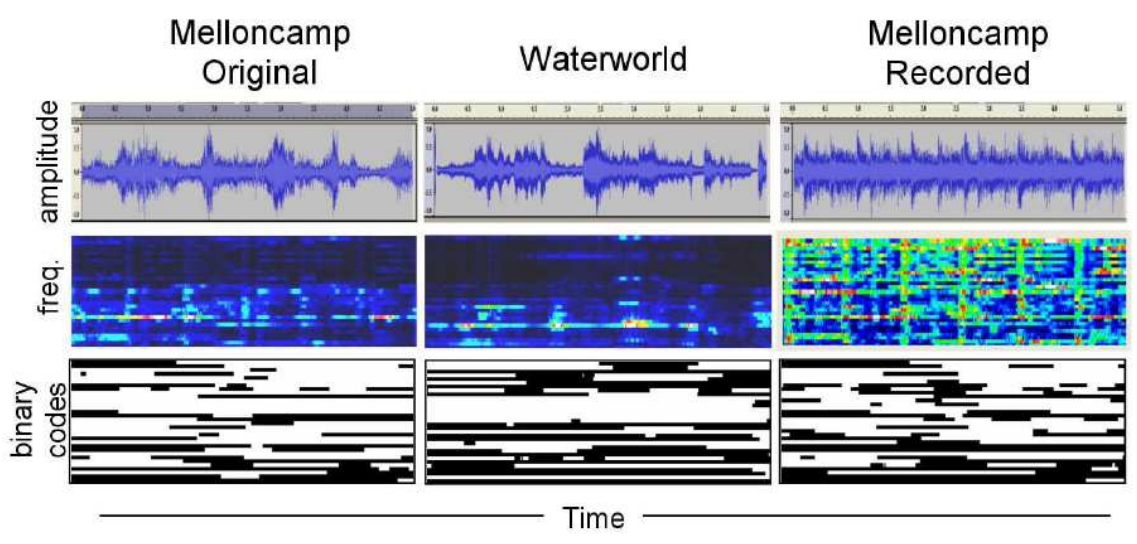
Cada filtre pot variar de banda entre 1 i 33, amplada de banda entre 1 i 33, i en temps entre 1 frame (11,6 ms) fins a 82 frames (951 ms) en passos exponencials de 1,5. Això produeix una quantitat de 25.000 filtres candidats. Seleccionem els M filtres discriminants i els seus corresponents llindars, creant un vector de M bits. Aquests vectors representaran segments d'àudio solapats. Aquest vector, o descriptor, pot ser computat ràpidament fent servir imatges integrals, i és suficientment estable com per a permetre fer hashing directe a la base de dades.

Una imatge integral I és una representació intermèdia de la imatge, i conté la suma dels grisos dels píxels de la imatge N, amb alçada Y i amplada X:

$$I(x,y) = \sum_{x'=0}^{x} \sum_{y'=0}^{y} N(x',y')$$

Aquesta representació intermèdia I (x, y) permet el càlcul de diferents característiques de manera ràpida. [7]

Un únic descriptor no conté la informació suficient com per a identificar eficientment el so d'una base de dades amb milers de sons. El que es fa és calcular descriptors per a finestres d'àudio solapades cada 11,6 ms. És a dir, per un fragment d'àudio d'uns 10 segons de duració faran falta 860 descriptors.



Aquesta figura mostra la potència i robustesa del mètode. Si es compara la cançó "John Melloncamp" original i el gravat a mala qualitat amb un micròfon mitjançant la resposta en amplitud sembla que no tinguin res a veure. Inclús si es compara amb el seu espectrograma, tot i ser més similar, resulta difícil trobar-ne alguna relació. En canvi, aplicats els filtres corresponents veiem que la seva codificació binària és molt semblant, fent que si sigui possible identificar aquesta darrera imatge.

L'objectiu és construir una representació on el segment de so original i les seves versions distorsionades puguin generar descriptors molt similars. A la vegada, els segments de so diferents han de produir descriptors diferents. Es fa servir l'algoritme anomenat P*airwise Boosting*.

2.1.1.1 Pairwise Boosting

L'objectiu de l'algoritme és construir un descriptor que ens permeti determinar la probabilitat de que dues mostres de so potencialment distorsionades pertanyin al mateix so. Formalment, això significarà entrenar un classificador $H(x_1, x_2) \rightarrow y = \{-1,1\}$, on x_1 i x_2 son dos espectrogrames diferents, i el valor y denota si les imatges provenen del mateix so ($y = 1$) o no ($y = -1$). El nostre classificador és un conjunt de M classificadors dèbils, $h_m(x_1, x_2)$, cadascun associat a una confidència c_m. Els nostres classificadors dèbils estaran compostos per un filtre f_m i un llindar t_m, tal que $h_m(x_1, x_2) = \text{sgn}[(f_m(x_1) - t_m)(f_m(x_2) - t_m)]$ on sgn és la signatura (teoria de grups). És a dir, si dos exemples generen valors de resposta del filtre en el mateix costat del llindar, aquestes seran etiquetades pel classificador dèbil com a derivats del mateix fragment de so. En cas contrari, seran etiquetades com a fragments diferents de so.

Una vegada que els nostres classificadors dèbils hagin estat sotmesos a un aprenentatge, qualsevol espectrograma *x* podrà ser transformat en un vector de M bits, permetent una indexació ràpida a través de tècniques de hashing.

Una tècnica per a entrenar aquests diferents classificadors podria ser la següent: de manera iterativa, els diferents classificadors son entrenats, i tots els pesos de les dades son modificats. És la tècnica utilitzada en l'algoritme *Adaboost*. Aquesta tècnica produirà resultats pobres per la raó següent: cap classificador dèbil pot actuar millor que la probabilitat, en promig, dels exemples que no pertanyen a la mateixa classe. Suposem que tenim una variable *x* aleatòria d'una distribució *D*, un filtre f_m i un llindar t_m, tal que $P(f_m(x) < t_m) = p$, on $0 \le p \le 1$. Si independentment i aleatòriament escollim dos exemples que no pertanyin a la mateixa classe, x_1 i x_2 del conjunt *D*, aleshores, la probabilitat de que x_1 i x_2 caiguin en bandes diferents de t_m ve donada per: $P(h_m(x_1, x_2) = -1) = 2p(1-p) \le 0.5$

Un parell d'exemples que no pertanyin a la mateixa classe ($y = -1$) incorrectament seran classificats en la mateixa classe si com a mínim la meitat del temps pertanyien a la mateixa classe per una mida de mostra de so suficientment gran. Aquest problema és solucionat fent servir l'algoritme asimètric *pairwise boosting*, on només en els parells que pertanyin a la mateixa classe seran recalculats els seus pesos. Els segments que no coincideixin seran normalitzats fent que la suma de tots ells sigui igual a ½. A continuació mostrem detalladament l'algoritme:

Pairwise Boosting

Entrada: Seqüència de n exemples

$< (x_{11}, x_{21}) > .. < (x_{1n}, x_{2n}) >$, cadascun amb la seva etiqueta $y_i \in \{-1,1\}$

Incialitzar: $w_i = \frac{1}{n}, i = 1..n$

Per m=1..M

1. Trobar la hipòtesi $h_m(x_1, x_2)$ que minimitzi l'error ponderat sobre la distribució *w*, on $h_m(x_1, x_2) = \text{sgn}[(f_m(x_1) - t_m)(f_m(x_2) - t_m)]$ per un filtre f_m i un llindar t_m

2. Calcular l'error ponderat: $err_m = \sum_{i=1}^{n} w_1 \cdot \delta(h_m(x_{1i}, x_{2i}) \neq y_i)$.

3. Assignar la confidència a $h_m : c_m = \left(\log(\frac{1 - err_m}{err_m} \right)$

4. Actualitzar els pesos pels parells que pertanyin a la mateixa classe:

   si $y_i = 1$ i $h_m(x_{1i}, x_{2,i}) \neq y_i$ aleshores

   $$w_i \leftarrow w_i \cdot \exp[c_m]$$

5. Normalitzar els pesos tal que $\sum_{i:y_i=-1}^{n} w_i = \sum_{i:y_i=1}^{n} w_i = \frac{1}{2}$

Hipòtesi final:

$H(x_1, x_2) = \text{sgn}(\sum_{m=1}^{M} c_m h_m(x_1, x_2))$

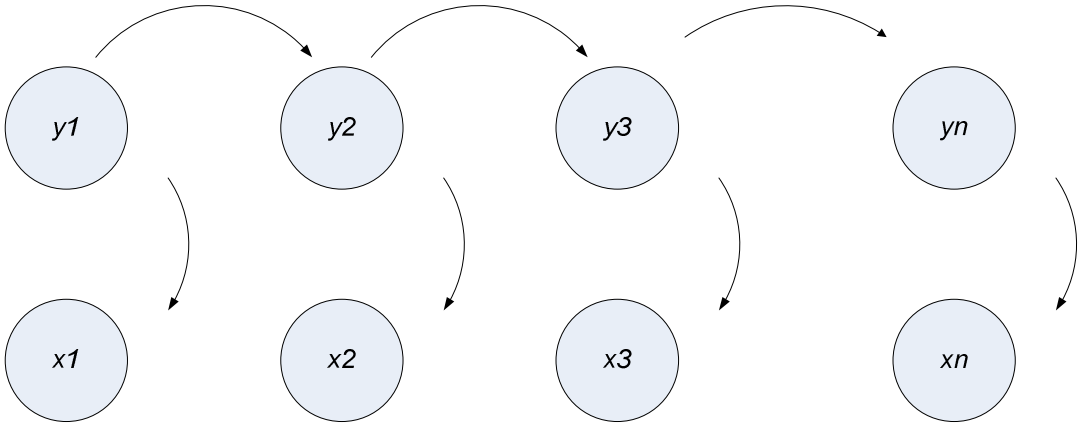
Algoritme de *Pairwise Boosting* per a aprendre una hipòtesi que determini si els membres de la parella x_1 i x_2 pertanyen a la mateixa classe, o bé a classes diferents. Aquest algoritme és asimètric perquè només els exemples que pertanyin a la mateixa classe son augmentats.

Aquest conjunt de 32 filtres als quals hem sotmès a l'aprenentatge esmentat anteriorment millora notablement els descriptors desenvolupats per Haitsma i Kalker per la identificació de música. Aquests darrers calculen la diferència entre freqüències veïnes en temps veïns.

2.1.1.2 Mètode del model d'oclusió

Aquest mètode ens permetrà determinar si dues parelles coincideixen basant-nos en la signatura, que estarà composta d'una sèrie de descriptors que tenen en compte el fet que algunes parts del so poden estar afectades per distorsions i soroll. Assumim que la probabilitat que un fragment gravat pertanyi a la mateixa classe que un fragment original dependrà de les diferències dels bits del descriptor en cadascuna de les signatures. Un fragment d'una versió distorsionada d'un so de la base de dades és poc probable que sigui completament atribuït a interferències, així com alguns descriptors son molt més probables de ser generats per una distorsió del so original, més que un soroll de fons.

El que es fa servir és un model com el següent:



Aquest model determina si un descriptor es un so de la nostra base de dades o algun altre so no identificat, el que anomenem "oclusió".

Un nivell per sota d'aquest model, assumim que la probabilitat de que un descriptor sigui generat per una oclusió depèn nomes de les dades i del descriptor anterior (en el temps) que va ser generat per una oclusió.

Més formalment, tenim un signatura $x^r = (x_1^r, x_2^r, ..., x_n^r)$ composta per *n* descriptors que son calculats d'un fragment de so distorsionat. Modelem la probabilitat de que la signatura sigui generada per una distorsió d'un fragment de so original amb signatura $x^o = (x_1^o, x_2^o, ..., x_n^o)$ de la següent manera:

$$P(x^r \mid x^o) = P(x^{r-o}) = \prod_{i=1}^{n} P(x_i^{r-o} \mid y_i) P(y_i \mid y_{i-1})$$

x_i^{r-o} denota la diferència dels bits entre els descriptors de la mostra amb soroll o distorsions (x_i^r) i la mostra original(x_i^o). $y_i = 1$ si el descriptor x_i^r prové d'una distorsió de l'arxiu de so original, i $y_i = 0$ si el descriptor prové d'una oclusió que ofusca el fragment de so original. La diferència dels descriptors $x_i^{r-o} \in \{0,1\}^M$ és un vector de M bits que denota si les sortides dels filtres amb llindar del descriptor original i del distorsionat tenen el mateix valor. Es modela la distribució x_i^{r-o} com el producte de variables aleatòries de *Bernoulli* independents i no idènticament distribuïdes. En aquesta implementació destinada al reconeixement de música, s'ha escollit M = 32, ja que hi han 66 paràmetres a estimar: 32 paràmetres de *Bernoulli* per cada $P(x_i^{r-o} \mid y_i = 0)$ i

$P(x_i^{r-o} \mid y_i = 1)$, i un paràmetre de transició de *Bernoulli* per cadascun dels $P(y_i \mid y_{i-1} = 0)$ i $P(y_i \mid y_{i-1} = 1)$.

Així doncs, en les nostres dades d'entrenament, nosaltres no sabrem si un descriptor particular és generat per música o bé per soroll. Necessitem un mètode per estimar les etiquetes de les dades y_i i els paràmetres. Es fa servir l'algoritme EM. Aquest algoritme es troba explicat en els documents annexos a aquesta memòria. En aquest model, els passos E i M de l'algoritme son senzills de realitzar degut a que la funció és fàcilment derivable.

Per una signatura $x^r = (x_1^r, x_2^r, ..., x_n^r)$, hem de trobar una signatura $x^o = (x_1^o, x_2^o, ..., x_n^o)$ a la base de dades que maximitzi $P(x^r \mid x^o)$. Finalment, es decideix si la signatura coincideix amb la més semblant a la base de dades de la següent manera: $P(x^r \mid x^o) > T$, on el llindar *T* controla la precisió i la sensibilitat .

2.1.1.3 Model Hash

Usant la representació descrita anteriorment, construirem signatures de tots els sons de la base de dades. Durant la obtenció de les dades, farem una cerca de similituds de cadascun dels descriptors que contingui la base de dades. La gran mida de la base de dades i el gran nombre de consultes requerides per cada fragment fa necessari trobar mètodes eficients de cerca per similitud en un espai de 32 bits (la longitud del descriptor).

Per a solucionar aquest problema es fa servir la tècnica basada en *locality-sensitive hashing* (LSH), una tècnica que ens permet fer cerques aproximades per similituds en un temps menor que el lineal, en la que encaixa molt bé la mètrica de la distància de *Hamming*. El que es fa és el següent; Es fa *hash* amb totes les signatures en una taula *hash* estàndard. Es defineix quins descriptors dins de la distància de *Hamming* de 2 del descriptor d'entrada son els veïns propers. Aquesta aproximació és significativament més ràpida que el LSH. S'ha observat que la distància de *Hamming* sense pesos enlloc del classificador com a base de la similitud dels descriptors és una aproximació raonable, ja que s'ha trobat que els valors de confiança per diferents classificadors son gairebé iguals que realitzant aquesta aproximació.

Una vegada s'han detectat tots els veïns propers, farà falta seleccionar un únic veí d'aquest conjunt. En comptes d'utilitzar una tècnica basada en el nombre de coincidències, el que es fa és utilitzar una verificació geomètrica . Per cada so candidat, es determina quins d'aquests descriptors son consistents a través del temps. Per a realitzar aquesta tasca, es fa servir l'algoritme RANSAC per a iterar a través dels diferents alineaments de temps candidats, la puntuació EM, la probabilitat de que la signatura consultada sigui generada per la signatura candidata, i la mètrica de la distància. L'algoritme RANSAC està explicat com a document annex a aquesta memòria.

Es fa la assumpció següent; la consulta pot ser alineada amb l'original una vegada un sol paràmetre (desplaçament temporal) ha estat determinat. El conjunt mínim és un sol parell de descriptors que pertanyen a la mateixa classe.

A la pràctica, s'ha trobat que aquest model convergeix en menys de 500 iteracions inclús amb la presència d'una oclusió molt significativa.

Una vegada tots els candidats rebuts de la base de dades han estat alineats, es selecciona el so guanyador amb la millor puntuació obtinguda de l'algoritme EM, assumint que passa el llindar mínim.

2.1.2 Aplicacions

Aquest mètode d'identificació de sons ha estat explotat per l'equip de recerca de Google per a aconseguir una interacció entre la televisió i l'ordinador. Google pretén crear un sistema de personalització Web a partir del so identificat en temps real [2].

Un dels grans avantatges d'aquest mètode és que no requereix una connexió directa entre l'ordinador i la televisió, ja que amb el so ambiental gravat a través d'un micròfon qualsevol a través de l'ordinador n'hi ha més que suficient.

Per altra banda, tampoc compromet la intimitat i privacitat dels usuaris, ja que a partir dels descriptors comentats anteriorment és impossible reconstruir el so inicial.

Es proposen un grup d'aplicacions interessants a partir d'aquest mètode d'identificació d'àudio.

2.1.2.1 Capes d'informació personalitzada

Aplicacions que donen informació addicional sobre el canal de *Mass Media* (com ara la televisió). En funció de la informació recollida a través de l'àudio ens mostraran informació relacionada, com ara moda, política, negocis, salut o turisme.

2.1.2.2 Comunitats Ad-Hoc

Amb la popularitat adquirida pels xats i els fòrums de discussions a través d'Internet és interessant explotar aquesta via. L'aplicació proposada consisteix en crear grups de discussió per a comentar els continguts de la televisió, com ara la nostra sèrie preferida. Es crearan comunitats en les que les estadístiques de so del conjunt d'usuaris coincideixi amb la base de dades d'identificació de sons.

2.1.2.3 Control d'audiència i popularitat

Aquesta és una aplicació força interessant, ja que les audiències son les que marquen les decisions en el món de la televisió. Actualment, captar l'audiència d'un programa requereix hardware addicional i una cooperació de l'usuari. Fent servir aquesta tècnica ens estalviem aquests problemes, i a més a més, podem saber en temps real la audiència més o menys precisa del programa en qüestió.

2.1.2.4 Preferits de vídeo

Si estem mirant un programa que ens pot interessar tornar-lo a veure en un futur, només hem de polsar un botó del nostre ordinador, i guardarem als nostres preferits el punt exacte del vídeo. Aquest preferit pot ser usat més tard per a obtenir el programa a través d'Internet, o bé per a compartir-lo amb les nostres amistats.

2.1.2.5 Investigació a Google Research

Des de febrer del 2006 existeix un *blog* de Google dedicat als temes de recerca als que ells s'estan dedicant [8]. Podem trobar escrits i texts relacionats amb la escalabilitat, algorítmica en general, aprenentatge de màquines, intel·ligència artificial i inclús processament de so. Aquests articles ens recorden que Google està immers en multitud de projectes relacionats amb l'àudio. Trobem certs documents [9] [10] que continuen investigant en el fet d'utilitzar l'espectrograma amb tècniques de visió per computador. Aquests documents mostren com tècniques de processament de visió per computador

combinades amb processament de *streams* de dades, poden crear un sistema eficient per a reconèixer sons degradats amb grans quantitats de soroll. Altres documents [11] mostren com tècniques de modelatge acústic usades sovint en reconeixement de la parla, i transductors d'estats finits usats per a representar i buscar en llargs grafs, poden ser usats en el problema de la identificació musical. Amb aquest experiment, el que es fa és un sistema que aprèn un alfabet comú de sons musicals, als quals anomenen *Music-phones*, que representen un gran conjunt de cançons mitjançant un graf, força gros per cert, on es possible aplicar una cerca eficient.

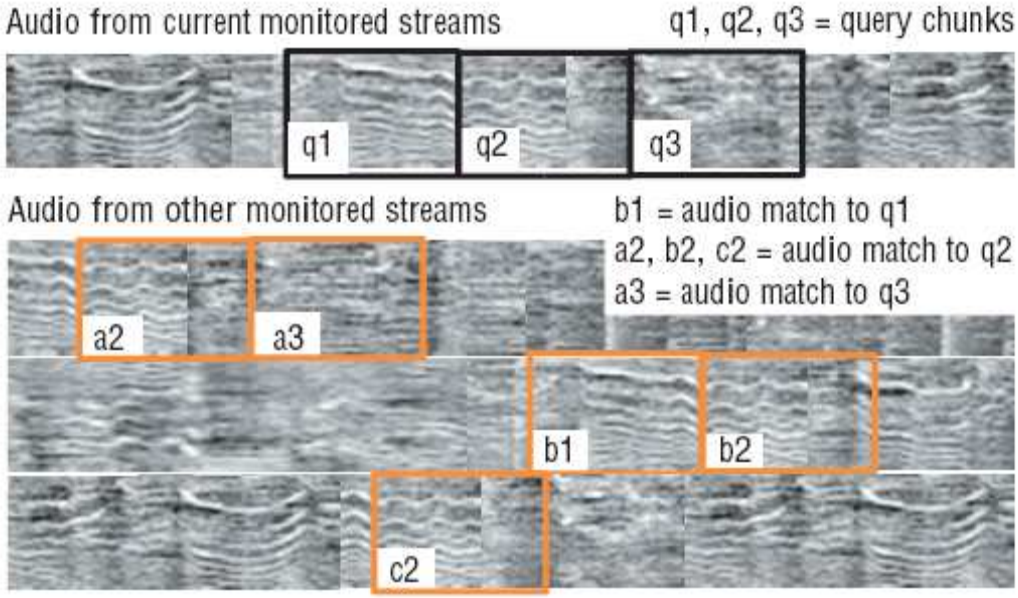
Una altra aplicació força interessant explicada en aquest *blog* és una, la qual va ser presentada a la *International Conference on Artificial Intelligence*, que descriu un sistema que aprèn les similituds rellevants en senyals musicals, i per altra banda manté la eficiència usant aquests models apresos mitjançant l'aprenentatge per a crear funcions *hash* a mida [12].

2.1.2.6 Detecció d'anuncis i publicitat

Combinant aquesta tècnica amb característiques de les imatges del vídeo se'ns proposa una altra aplicació, que consisteix en la detecció d'anuncis en *streams* de vídeo [4].

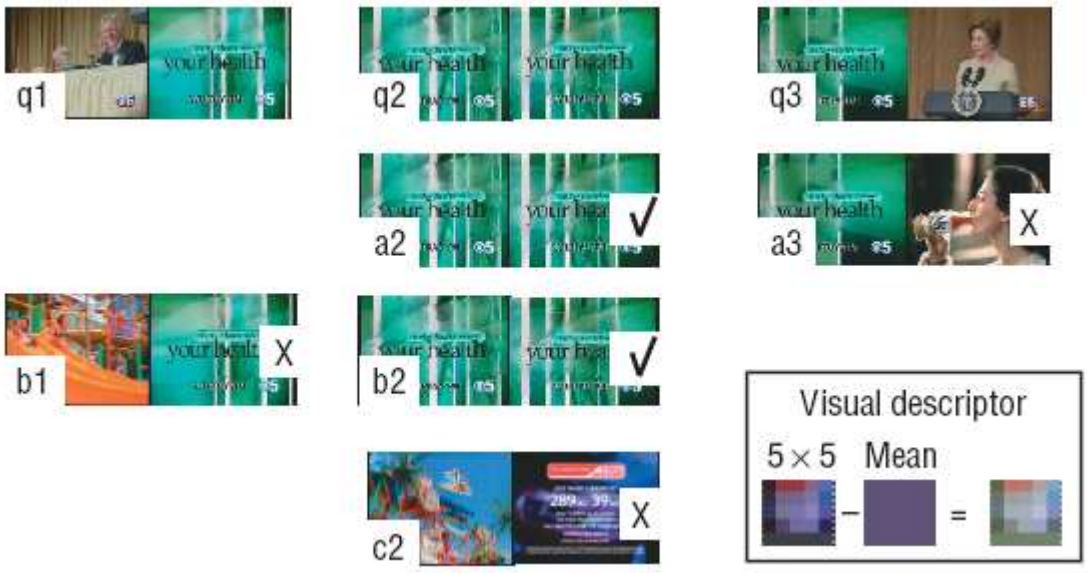
Aquest procés de detecció d'anuncis consta de 3 fases:

La primera fase consisteix en relacionar els *frames* d'àudio que es repeteixen. És a dir, buscar parts de so que es repeteixin al llarg del temps. Això es fa mitjançant la tècnica explicada anteriorment, però sense fer servir fragments que se solapen. Cadascun d'aquests fragments té una duració de 5 segons, que aproximadament és la meitat de la duració de l'anunci més curt.



Primera fase: Detecció dels fragments que es repeteixin.

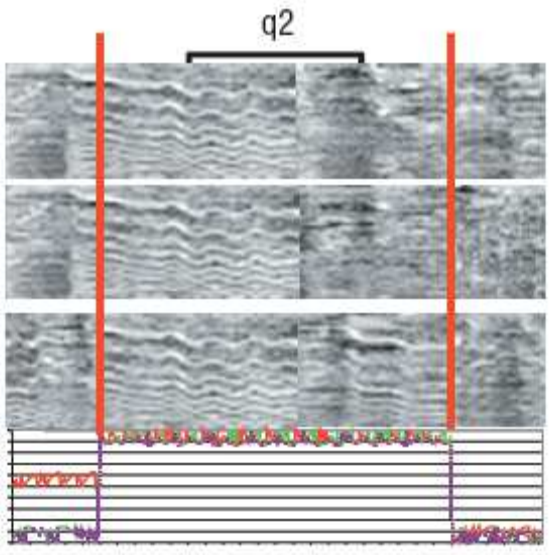
La segona fase consisteix en verificar les relacions anteriors mitjançant pistes visuals, ja que en fase anterior es poden produir molts falsos positius. El que es fa és agafar imatges de 5x5 RGB de 24 bits, agafant-ne 3 per segon. A partir d'aquestes imatges es calcula la mitjana dels colors. Es produeix una normalització d'aquestes imatges per a fer-les coincidir en la base de dades, i a partir d'aquest punt passem a la última fase d'aquest procés.



Segona fase: Validació dels candidats detectats mitjançant la pista visual de la mitjana aritmètica dels

colors de la imatge.

La tercera fase fa la hipòtesi de que totes les relacions anteriors que han passat les proves de consistència, tant de so com les visuals, són parts d'anuncis, i per tant, encerts reals. Durant la última fase el que es fa és detectar realment el principi i el final de l'anunci. Recordem que teníem deteccions de 5 segons de duració. Aquest últim punt es basa en tècniques estadístiques i heurístiques, aconseguint una taxa d'encert entre el 95 % i el 99 %.



Tercera fase: Refinar la segmentació temporal a 11 ms de resolució, i detectar exactament el inici i el final de l'anunci

En definitiva, amb la tècnica descrita anteriorment per la detecció de sons mitjançant l'espectrograma i visió per computador, combinada amb altres tècniques més o menys innovadores, és possible crear un conjunt nou d'aplicacions que poden servir en diferents camps de la ciència i entreteniment.

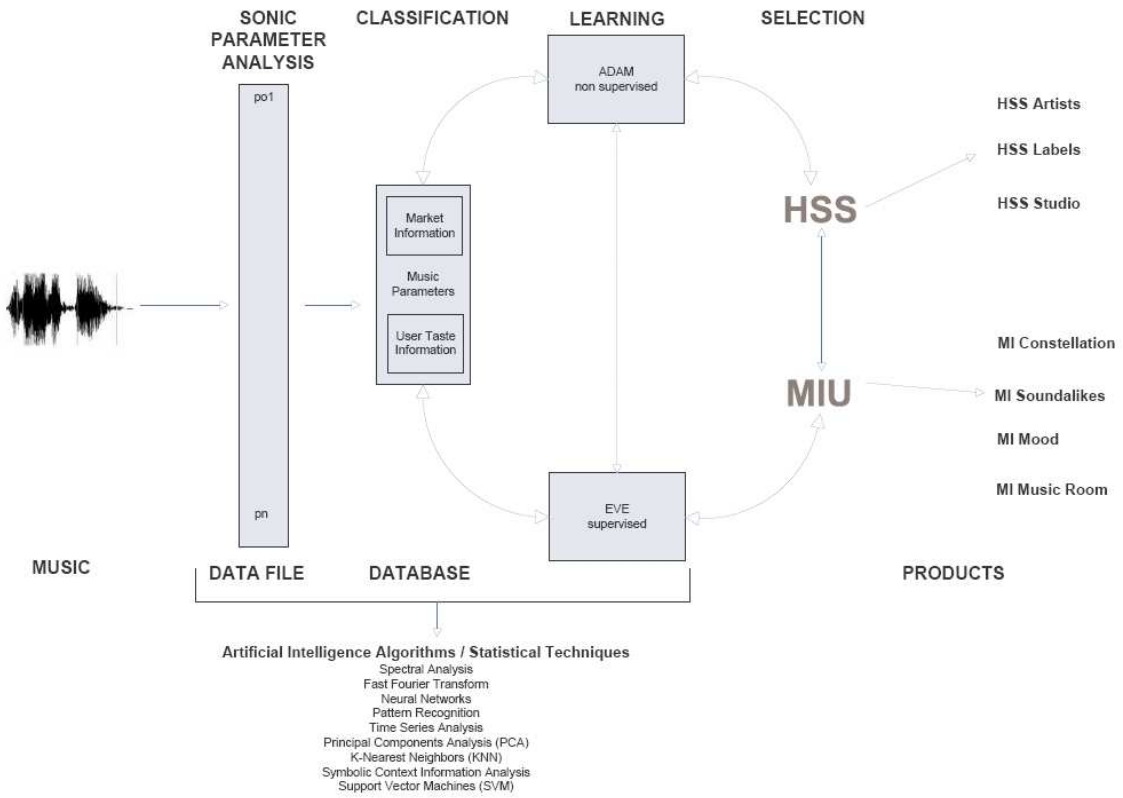
2.2 Hit Song Science

El món de la indústria discogràfica és un món que mou una gran quantitat de diners. Aquest és un tema que no se li escapa a ningú, i combinant tècniques d'intel·ligència artificial amb altres tècniques de tractament d'àudio, es poden arribar a crear aplicacions molt interessants per aquesta indústria. És el cas de *Poliphonic HMI*, que ha desenvolupat un programa informàtic pensat per ser capaç de determinar, de forma matemàtica, si una cançó compleix els requisits per a convertir-se en un *Hit* [13] .

Aquesta companyia treballa amb grans multinacionals, com ara *Sony-BMG*, *Universal*, *Emi* o *Warner*.

El que fan és aïllar certs aspectes de la cançó, com ara la melodia, la harmonia, el tempo, el to, les octaves, el ritme o la progressió d'acords. Aquest procés l'anomenen *Deconvolució Espectral*. Cada cançó és situada en un punt d'una reixa n-dimensional, que ells anomenen l'univers de la música, on les cançons amb similituds matemàtiques són posicionades de manera propera. A partir d'això, es determinen quines són les posicions d'aquest univers musical que formen part de la cançó èxit, i se li assigna una puntuació a cada cançó.

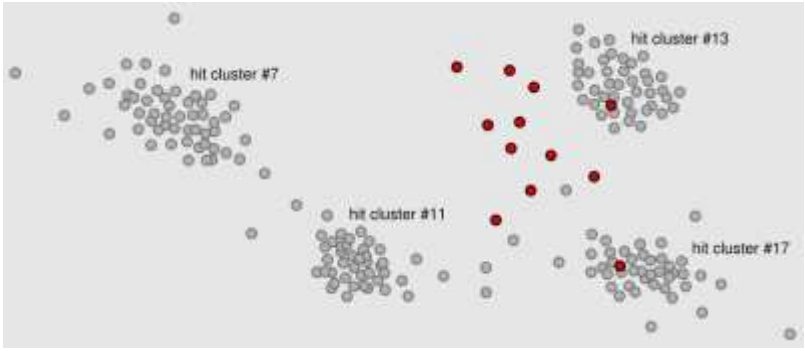
La següent imatge il·lustra el procés que segueix aquesta tecnologia:



La senyal musical d'entrada és sotmesa a una sèrie de processos de tal manera que s'extreuen les característiques principals de la cançó, com ara l'harmonia o el tempo mitjançant tècniques com ara l'anàlisi espectral i la transformada de *Fourier*. Aquests paràmetres serviran d'entrada a un classificador que utilitza varies tècniques simultàniament, com ara xarxes neurals, *PCA*'s i K–veïns més propers. Aquest classificador és entrenat mitjançant tècniques d'aprenentatge supervisat i no supervisat.

La tècnica que ells fan servir recorda en gran mesura a la que nosaltres hem fet servir. Tot es basa en el mateix concepte simple: A partir d'una senyal de so d'entrada

n'extreuen unes característiques determinades. Aquestes característiques diferencien una cançó èxit d'una que serà un fracàs. A partir d'aquestes dades, mitjançant tècniques com podrien ser les basades en la distància euclidiana, o bé classificadors, s'obté el resultat de la cançó.



Aquest és un exemple de la tècnica del *Hit Song Science*. Existeixen diferents agrupacions que defineixen les cançons d'èxit. Aquestes tenen propietats matemàtiques semblants. Si la cançó a analitzar pertany a alguna d'aquestes agrupacions significarà que existeixen possibilitats de que la cançó sigui un èxit [13].

Una revista musical [14] va testejar aquest sistema amb un grapat de cançons en català molt conegudes, com ara *El far del Sud* de *Sopa de Cabra*, o *Bon Dia* dels *Pets*, obtenint una puntuació de 8,53 i 6,52 respectivament.

2.3 Speech Features

Altres recerques sobre el so ens situen a la Universitat de Massachusetts, concretament al *MIT Media Lab*. Aquest departament es dedica a la recerca sobre aplicacions en les que es produeix una convergència entre el món de la informàtica i les arts en general. Tenen obertes branques de recerca sobre la comunicació entre les persones, i sobretot la comunicació no verbal [3]. Han explotat un nou marc de treball que se centra en els senyals socials de la persona que parla, així com de la seva actitud o intencions a l'hora de comunicar-se a través de la parla.

Aquests estudis s'han basat en unes senyals que poden ser automatitzades amb un ordinador, és a dir, no cal que una persona faci la traducció ni etiquetatge de cap tipus. També són universals, no fa falta ser entrenat per a uns usuaris concrets. Aquestes senyals se les anomena Activitat i Èmfasis.

2.3.1 Comunicació social

La comunicació social per a senyals és el que percebem quan observem una conversa en un llenguatge que no ens és familiar i que desconeixem. Tot i així som capaços de veure qui porta el pes de la conversa, si s'estableix una connexió amistosa entre els parlants, o bé si la persona que escolta expressa alguna mena d'empatia.

El concepte important és el següent: La comunicació verbal no acaba amb el contingut sintàctic i semàntic de les paraules.

Les pistes no lingüístiques que la persona parlant fa servir per a guiar els oients són anomenades Entonació. La entonació inclou una sèrie de factors com ara el to de veu, el ritme o bé la sonoritat. Aquest tipus de comunicació pot passar de manera conscient, o bé inconscientment.

2.3.2 Plataforma d'anàlisi de la parla

2.3.2.1 Característiques

Aquests estudis han portat a la creació d'una plataforma d'anàlisi de la parla, que en mesura certes característiques prosòdiques de manera ràpida i eficient. Aquestes característiques estan basades en parla sonora, és a dir, segments de parla dels quals el seu espectre mostra una estructura harmònica forta, és a dir, existeix presència de vocals.

Aquesta plataforma funciona de la següent manera:

Durant la fase inicial s'extreu un conjunt bàsic de característiques de la parla d'un arxiu d'àudio amb una freqüència de mostreig de 8000 Hz. El processament es fa amb una finestra de 256 (32 ms) amb una mida de passes de 128 (16 ms). Se'n extreuen les següents mesures:

- f_0: Freqüència fonamental, Bàsicament consisteix en el to de la veu.
- Entropia espectral: Mesura la aleatorietat del segment en el domini freqüencial.
- Autocorrelació espectral: Autocorrelació de la transformada de *Fourier*. Un segment amb veu tindrà forts pics degut a la seva periodicitat.
- Energia: El volum (o sonoritat) del segment.
- Energia d/dt : La derivada de la energia.

A partir d'aquestes característiques bàsiques s'apliquen tècniques d'anàlisi per a determinar en quins segments hi ha gent parlant, i quins poden ser agrupats per a constituir una frase. La comunicació no verbal és extreta dels segments de parla, que corresponen als sons vocals. La tonalitat de la veu emfatitza l'aspecte melòdic i la importància de les vocals.

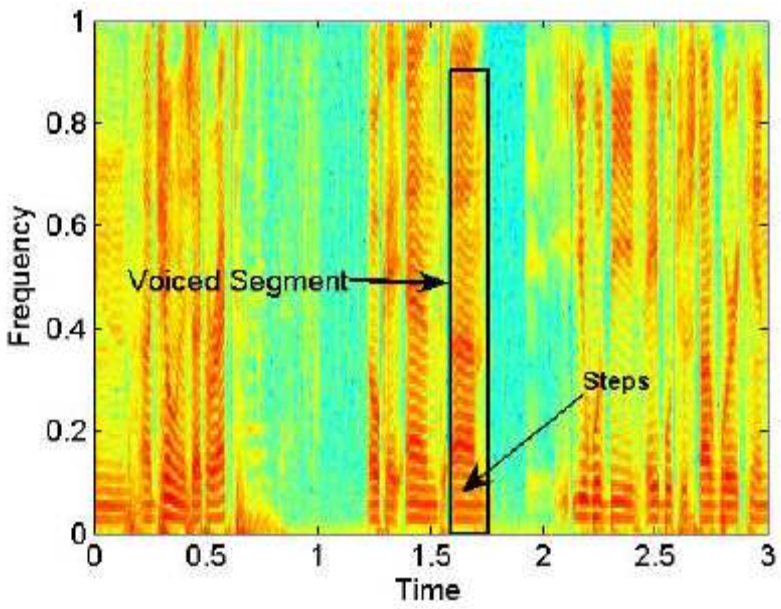
Una vegada s'ha realitzat aquest anàlisi, se'n extreuen una sèrie de característiques a més alt nivell per a un període de temps seleccionat. Són les següents:

- Mida del segment de veu: La duració del so vocal. Bàsicament ens indica la mida de cada síl·laba de la parla.
- Mida del segment de parla: La duració de la frase.
- Fracció de temps parlat: Percentatge del total temps entre frases.
- *Voicing Rate*: El nombre de segments amb veu (bàsicament síl·labes) per unitat de temps.
- Entropia de la duració de segments de parla: Mesura de l'aleatorietat en les mides de les frases.
- Entropia de la duració de les pauses: Mesura de l'aleatorietat en les mides de les pauses entre frases.
- Duració: Duració de la sessió (en segons).

2.3.2.2 Detecció de la parla

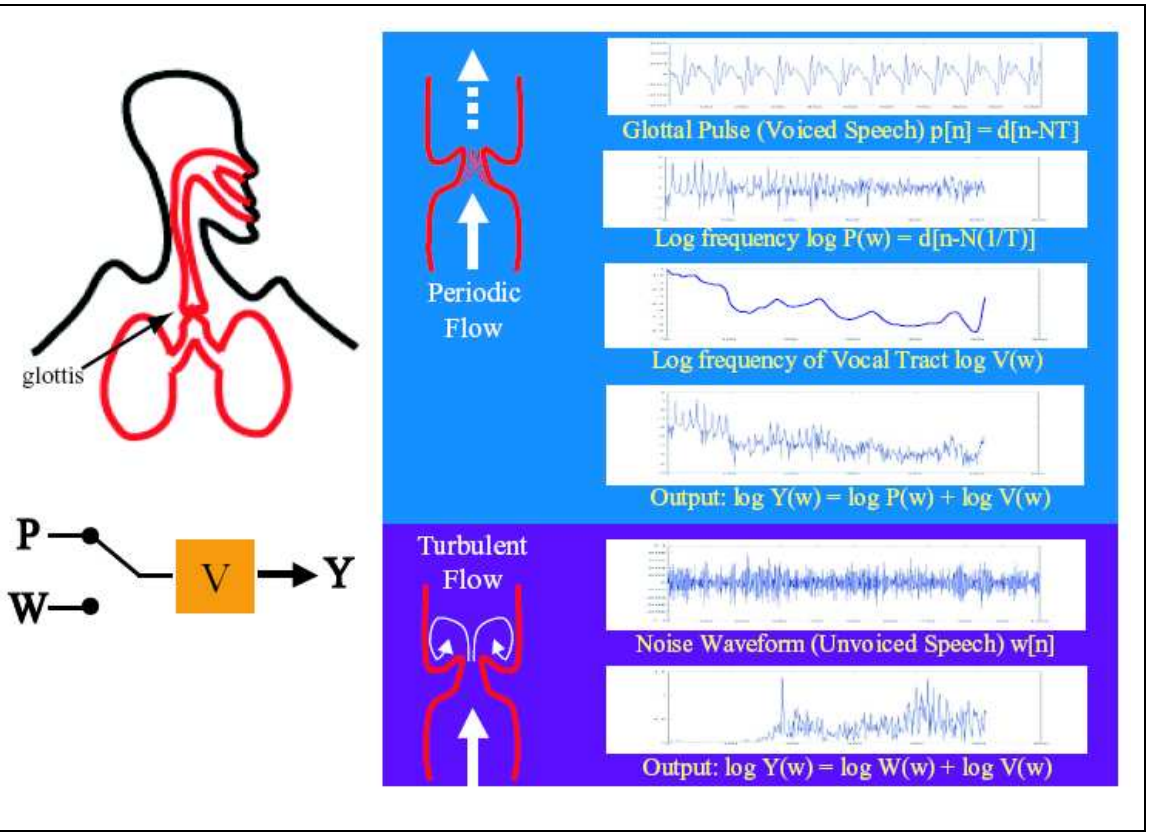
Una part important d'aquest treball consisteix en la detecció de la parla[20] [21]. S'ha realitzat de la següent manera.

Si observem un espectrograma, els fragments vocals poden ser identificats fàcilment:



La estructura formada a base d'esglaons és un indicador d'un segment vocal. Cada esglaó defineix una freqüència formant.

Per a comprendre el problema de detecció de parla i de veu, hem de comprendre el procés de producció de parla que efectua el cos humà. La següent il·lustració simplifica el funcionament del nostre cos:



Sistema de producció de parla. Els pulmons empenyen l'aire a través del glotis per a crear un pols periòdic, fent que es tanqui i s'obri, o bé mantenint-lo obert per a produir un flux d'aire turbulent. L'espectre periòdic, o bé pla, és reconvertit pel tracte vocal amb una funció de transferència *V(w)*.

La parla pot ser dividida en dos grups. Els grups vocals i els consonants. Durant els segments vocals, els pulmons proporcionen una pressió d'aire cap al glotis, i a partir de cert punt s'obre i deixa sortir un pols d'aire. Immediatament després es torna a tancar. Això passa amb un període fixat, i el resultat és un tren de polsos *p[n]*. La seva transformada de *Fourier P[W]* és un tren de polsos amb un període que coincideix amb el to del senyal. Aquest tren de polsos viatja a través del tracte vocal, que filtra el so amb V[w] en el domini freqüencial. La sortida és el senyal Y[w] és la següent: $Y[w] = V[w] * P[w]$.

En el cas d'un so consonant, els pulmons aporten l'aire suficient al glotis per a mantenir-lo obert. El so és construït a partir de la configuració del tracte vocal, incloent-hi la posició de la llengua i la boca. El so que es genera com a sortida no és periòdic.

S'ha inclòs l'alfabet fonètic internacional com a document annex a aquesta memòria, on s'observen tots els sons que generem amb la parla. En aquest alfabet es veu clarament la separació que existeix entre sons consonants i sons vocals.

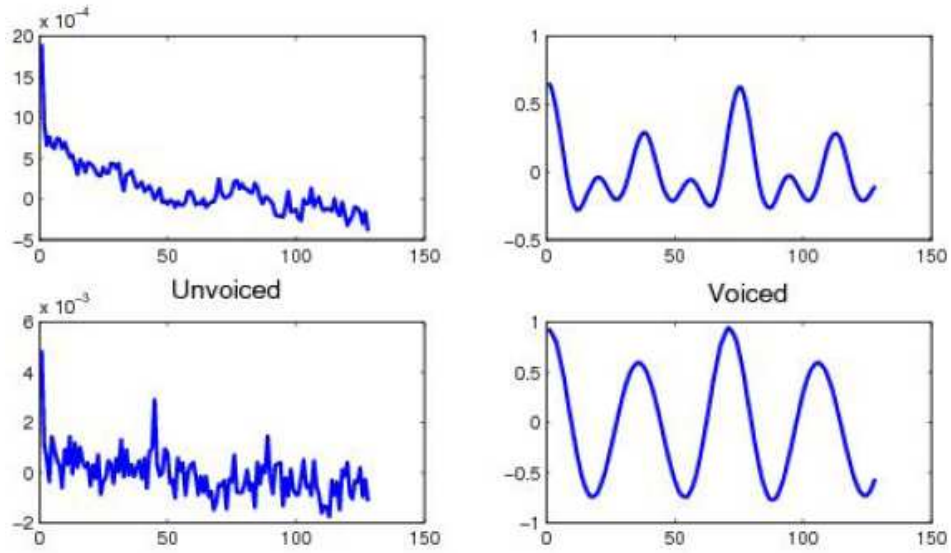
Per a la detecció d'un so vocal i un no vocal (o consonant), utilitza l'autocorrelació i la entropia espectral. A partir d'un model de *Markov* ocult, s'identifiquen els segments de so on existeix parla basant-se en els segments vocals i no vocals.

2.3.2.2.1 Autocorrelació

L'autocorrelació és calculada per a cada frame amb la següent fórmula:

$$A[k] = \frac{\sum_{n=k}^{N} s[n]s[n-k]}{\left(\sum_{n=0}^{N-k} s[n]^2\right)^{\frac{1}{2}}\left(\sum_{n=k}^{N} s[n]^2\right)^{\frac{1}{2}}}$$

A continuació mostrem un gràfic de l'autocorrelació per a dos fragments amb sons vocals i dos més amb sons no vocals:



Els fragments amb sons vocals mostren pocs pics, però molt forts. Això és degut a que la senyal és periòdica. En canvi, els fragments sense sons vocals tenen molts pics, però dèbils. D'aquí es pot deduir que el valor del pic màxim i el nombre de pics son bons indicadors per a saber en quina classe de segment ens trobem.

Desafortunadament, una senyal amb valors molt petits i periòdica també produirà grans pics. Per a solucionar aquest problema, el que es fa és afegir la entropia espectral al model.

2.3.2.2.2 Entropia espectral

Per a calcular la entropia d'un *pitch* (freqüència de so; El *pitch* que percebem d'un so és la resposta de l'aparell auditiu a la freqüència.) *P[w]* fa falta sotmetre'l prèviament a un procés de normalització: $p[w] = \frac{P[w]}{\sum P[w]}$

L'entropia relativa espectral és la divergència KL entre *p[w]* i la mitjana local de l'espectre *m[w]*: $H_{rel\_spec} = -\sum_{w} p[w] \log \frac{p[w]}{m[w]}$

El *pitch* fa que els segments vocals estiguin molt ben estructurats, mentre que els segments no vocals seran molt més aleatoris.

2.3.2.2.3 Freqüència fonamental

La freqüència fonamental és la freqüència més baixa en una sèrie d'harmònics. És calculada basant-se en que els segments vocals mostren certes bandes que es repeteixen,

com hem vist anteriorment. Aquestes bandes que es repeteixen s'anomenen formants. Tots els formants son múltiples exactes de la freqüència fonamental. Per a extreure aquestes bandes fa falta aplicar la convolució a l'espectrograma amb una senyal periòdica, com ara el cosinus. La convolució es calcula amb la part real de la transformada ràpida de *Fourier*. El pic més baix amb una freqüència adequada(entre 20 i 500 Hz) correspon a la freqüència fonamental.

En termes de superposició de senyals sinusoïdals, com ara les sèries de *Fourier*, la freqüència fonamental es la freqüència més baixa del sumatori.

2.3.2.2.4 Energia

La energia, o sonoritat és una mesura subjectiva. Depèn del oient. Determina la intensitat amb la que un so és percebut per l'oïda humana.

Depèn de la intensitat del senyal, però també de la seva freqüència, amplitud, i altres variables, com poden ser la sensibilitat de l'oïda de la persona que escolta, o bé de la duració del so.

Com que no és una magnitud absoluta, el que es fa és mesurar el nivell de sonoritat. És a dir, determinar la sonoritat del so en relació a un altre. Es mesura en dos unitats; el fon (que és la utilitzada per aquesta plataforma d'anàlisi de parla) i el soni.

- Fon: és la sonoritat d'un so sinusoïdal de 1KHz amb un nivell d'intensitat de 0 dB_{SPL}:

$$S = 10 * \log_{10}\left(\frac{I}{I_0}\right)$$

- Soni: Aquesta unitat és capaç d'establir la relació real de la sonoritat de sons diferents. És la sonoritat d'un so sinusoïdal de 1KHz amb un nivell d'intensitat de 40 dB_{SPL}.

2.4 Les emocions a través del so

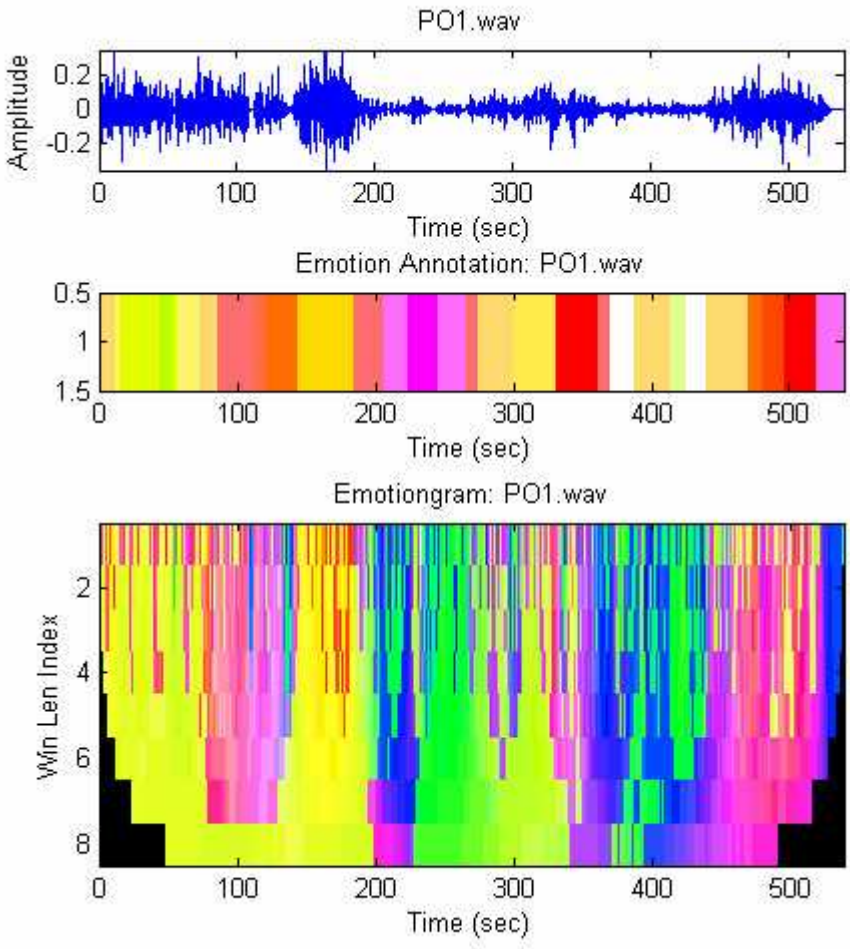
A vegades, si la cançó és realment bona, és capaç d'influir en els nostres sentiments, fins arribar al punt de fer-nos emocionar. Certs estudis intenten quantificar les respostes emocionals a través de la música [15]. Aquests estudis busquen la resposta a la següent pregunta: Per què algunes actuacions musicals ens fan emocionar i altres no? És a dir, pot ser quantificada una emoció? Han arribat a la conclusió de que no hi ha cap garantia

de que, una cançó que segueixi la fórmula de l'emoció extreta d'aquests estudis, no garanteix que realment aquesta cançó faci emocionar. La emoció és un sentiment molt complex i depèn de molts factors, la majoria no quantificables.

2.4.1 Emotiongram

El projecte *Emotiongram* [16] ha intentat extreure un conjunt de característiques de l'àudio per a la detecció d'emocions, i permetre una visualització en un gràfic en dos dimensions anomenat *Emotiongram*. El procés que se segueix és el següent:

El fitxer de so inicial és partit en finestres en funció de diferents temps, i les característiques principals rellevants per les emocions són extretes, escalades i combinades. Aquestes senyals corresponen a coordenades en l'espai de les emocions, i són mapejades en diferents colors de tal manera que cada color correspon a una emoció diferent.



Exemple d'*Emotiongram*. Correspon a la cançó *Piano Concerto No. 1* de *Tchaikovsky*.

Durant la realització d'aquest experiment, cada fitxer *.wav* és convertit en un midi mitjançant un software especialitzat en realitzar aquesta tasca, i mitjançant la *Midi Toolbox* de Matlab, se li realitzen una sèrie d'accions a partir de càlculs de diferents finestres fins a obtenir el *Emotiongram*.

IMPLEMENTACIÓ

En aquest apartat mostrarem les diferents aplicacions que hem realitzat, així com tota una sèrie d'experiments que hem realitzat amb elles.

3 Implementació

Tots els experiments i aplicacions que he realitzat han estat fets amb el sistema operatiu *GNU/Linux*. Les aplicacions són executades en la màquina virtual de Java, o bé en Matlab, pel que no descartem que puguin arribar a funcionar en altres sistemes operatius.

3.1 Preparació dels vídeos per a ser processats

La obtenció dels arxius de vídeo dels diferents telenotícies no ha estat fàcil. Hem tingut certs problemes tècnics que hem solucionat de la millor manera possible.

Inicialment, la idea era capturar aquests vídeos mitjançant una capturadora de TDT. La varem provar des de Manresa i des de Sabadell. A cap d'aquests dos llocs rebíem suficient senyal per a poder gravar les emissions en unes mínimes condicions. Per tant, varem haver de buscar una alternativa.

Finalment, la obtenció la varem realitzar amb un DVD gravador de senyal analògica. El problema és que el procés va ser molt lent i requeria un excessiu nombre de passos per a obtenir un arxiu de vídeo en un format acceptable per a treballar-hi.

3.2 Obtenció de la informació necessària en XML

La part central d'aquest projecte gira al voltant de la obtenció de les *Speech Features*. Amb elles serem capaços de diferenciar els diferents programes de notícies. Així doncs, el nostre primer pas consistirà en convertir els arxius de vídeo en arxius WAV per a ser analitzats.

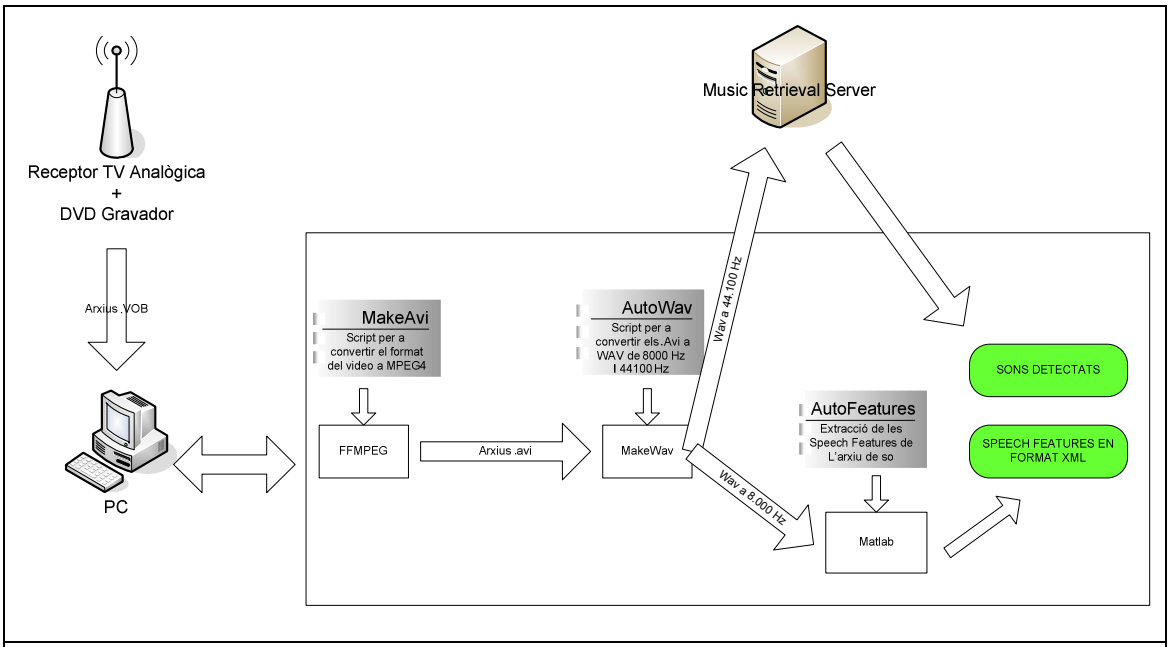
La *Toolbox* de les *Speech Features* treballa amb arxius WAV de 8000Hz. Així doncs, varem haver de reconvertir els arxius AVI a aquests darrers. Per agilitzar aquest procés varem crear un *Script* en *Bash* per a fer aquest procés de manera automàtica.

Una vegada hem obtingut aquests arxius WAV, el següent pas a seguir consisteix en analitzar els arxius amb la *Toolbox* de les *Speech Features*. Són una gran quantitat de dades, ja que obtenim 6.250 conjunts de característiques per minut. Això equival a 2,5 vegades més que el nombre de *frames* de vídeo de l'arxiu.

Segurament, aquesta ha estat la dificultat més gran en la que ens hem trobat a l'hora d'analitzar les dades, ja que treballem amb arxius d'una mida considerable (sobre l'hora de duració, i a vegades inclús una mica més).

Per altra banda, i mirant d'explotar una mica més la possibilitat que ens ofereix la tècnica desenvolupada per Yan Ke juntament amb altres investigadors de Google [1] hem creat la versió WAV a 44100 de l'arxiu de vídeo original per a poder aplicar la tècnica de detecció de sons mitjançant el seu espectrograma.

Com que és un procés una mica enrevessat, adjuntem el següent esquema per a fer més entenedor el procés realitzat:



Procés seguit en la conversió de l'arxiu inicial en diferents formats fins a l'estat final

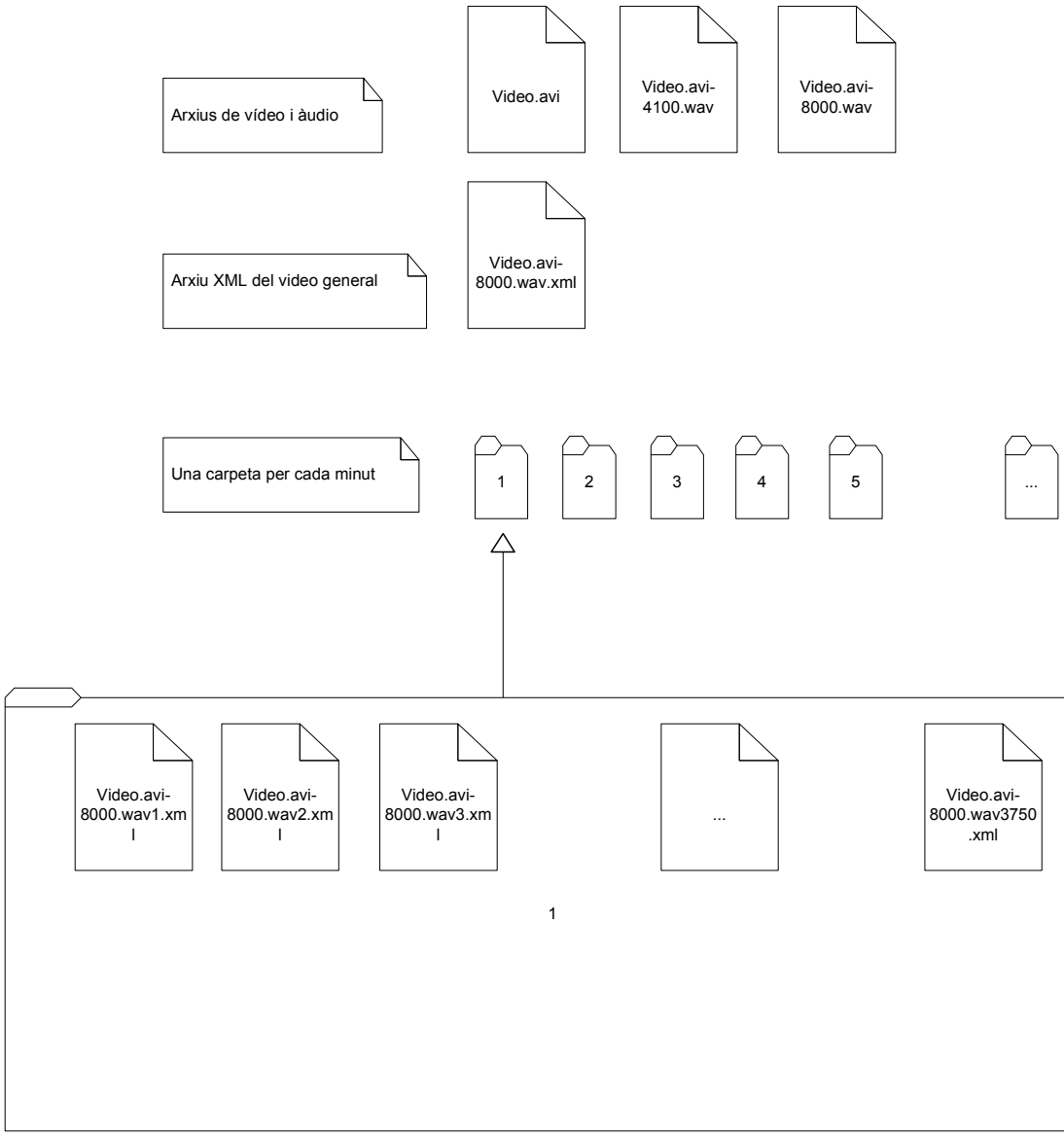
3.2.1 XMLGUI

XMLGUI és la primera aplicació creada per a la realització d'aquest projecte. Està realitzada en Matlab. És una aplicació que permet obrir arxius de so WAV i carregar-los en memòria, així com la visualització de la seva resposta en amplitud i del seu espectrograma.

La visualització del seu espectrograma ens ha interessat a mode de curiositat per a visualitzar les diferents imatges que comparará el servidor de detecció de sons (*Music Retrieval Server*).

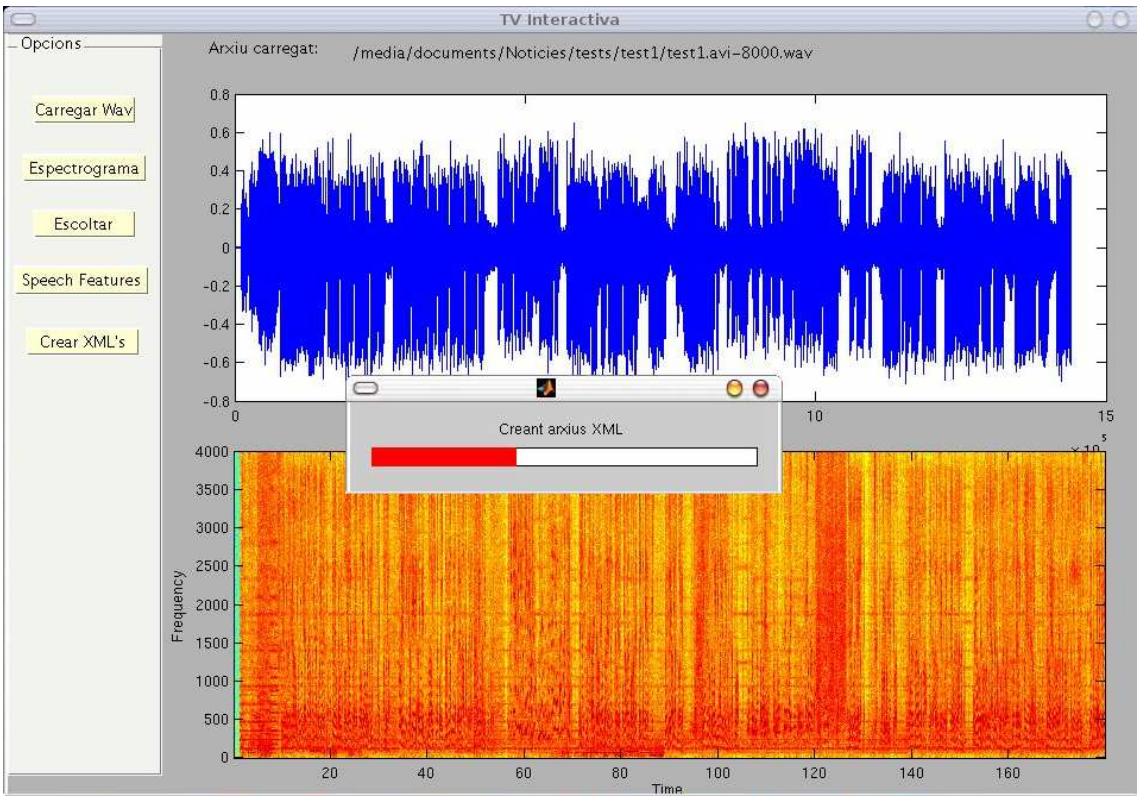
A part de poder visualitzar el senyal de so de diverses maneres, aquesta aplicació calcula les *Speech Features* [3] i n'extreu el resultat de cada frame en arxius XML, a més d'un arxiu resum de tot el vídeo analitzat .

Com hem comentat anteriorment, existeix una gran quantitat d'arxius XML per a cada vídeo d'una hora (al voltant d'uns 250.000). Aquesta quantitat tant gran d'arxius és impossible tractar-la en un sol directori. Així doncs, hem pres la decisió de crear una estructura jeràrquica de carpetes. Cada minut l'hem posat en una carpeta separada de l'anterior amb l'objectiu de no col·lapsar el sistema de fitxers del Sistema Operatiu. Un esquema d'aquesta organització és el següent:



Estructura dels arxius a analitzar.

D'aquesta manera, ja serà possible analitzar els arxius des d'una altra aplicació, ja sigui en C++, Java, o qualsevol altre llenguatge en el que sigui possible llegir arxius XML, que són la gran majoria.



Captura de pantalla de l'aplicació *XMLGUI*. S'observa a la part esquerra de la finestra les diferents opcions disponibles per a l'usuari. *Carregar Wav* ens permet seleccionar i obrir l'arxiu desitjat. *Espectrograma* dibuixa en el visor inferior l'espectrograma del senyal. *Escoltar* permet escoltar l'arxiu de so seleccionat. *Speech Features* calcula les característiques que volem extreure del senyal i les guarda en la memòria interna del programa. Finalment, *Crear XML's* exporta aquestes característiques calculades en el pas anterior en format XML per a poder-hi treballar més tard.

3.2.2 AutoFeatures

Aquesta aplicació, també escrita en Matlab, és pràcticament idèntica a la anterior, però aquesta no disposa d'entorn gràfic. Funciona únicament per línia de comandes. L'avantatge d'aquesta aplicació és que, a partir del directori que indiquem a la entrada del programa, busca de forma recurrent per les carpetes fins a explorar tots els fills de la carpeta inicial. És a dir, ens estalvia la feina d'indicar-li manualment quins són els fitxers que volem analitzar, ja que els busca automàticament. Força útil per a poder processar la gran quantitat de vídeos dels que disposem, ja que per ella mateixa ja és

una tasca prou lenta. Tret d'aquest detall, el funcionament d'aquesta aplicació és calcat a l'anterior, obtenint la mateixa sortida de fitxers.

Cada *frame* de les *Speech Features* conté una sèrie d'informació corresponent a aquell període de temps. Hi ha informació com ara la *formant frequency*, que bàsicament defineix el to de la veu, la energia del frame, que defineix la sonoritat o volum, a més d'altres dades com ara la entropia espectral, la derivada en funció del temps de la energia o els coeficients *Mel-Cepstrum*, que poden resultar prou útils a l'hora de reconèixer els diferents locutors. També conté informació sobre si s'ha detectat un so vocal o bé parla.

Cal destacar que a partir d'aquest punt, ja disposem de la informació necessària per a la realització dels nostres experiments i aplicacions derivades d'aquesta informació.

3.3 Reconeixement de sons.

Una vegada hem obtingut les dades necessàries per a la realització del nostre estudi, necessitem alguna eina per a comprovar que realment la *Toolbox* de les *Speech Features* funcioni bé, i que sigui capaç de diferenciar els estats on el presentador parla i no parla. Aquesta és la part més complicada de tot el procés de les *Speech Features*. Per tant, podem deduir que si aquesta part funciona bé, com que depèn de la resta, la resta també funcionarà bé.

Per altra banda, degut al nostre interès en el reconeixement de sons aplicant tècniques de visió per computador, decidim incorporar al nostre projecte el servidor de reconeixement de sons, del qual es pot descarregar lliurement el seu codi font des de la pàgina de l'autor [1].

Per tant, ens varem marcar com objectiu una aplicació que permetés explotar aquests dos conceptes. Per una part, visualitzar de manera gràfica en quin moment el presentador està parlant i visualitzar en quins instants del vídeo el detector de sons ha reconegut un so que té emmagatzemat a la base de dades. Escollim utilitzar els sons de la sintonia dels diferents telenotícies per aquest darrer objectiu.

3.3.1 PFC Player

Aquesta aplicació, realitzada amb Java, és l'encarregada de realitzar aquesta tasca. Disposa d'una interfície gràfica que permet obrir un arxiu de vídeo AVI, del qual

s'encarrega de buscar els seus arxius XML corresponents, emmagatzemats amb la estructura de carpetes anteriorment descrita, i crear una imatge, que és mostrada per pantalla mitjançant la interfície gràfica del programa.

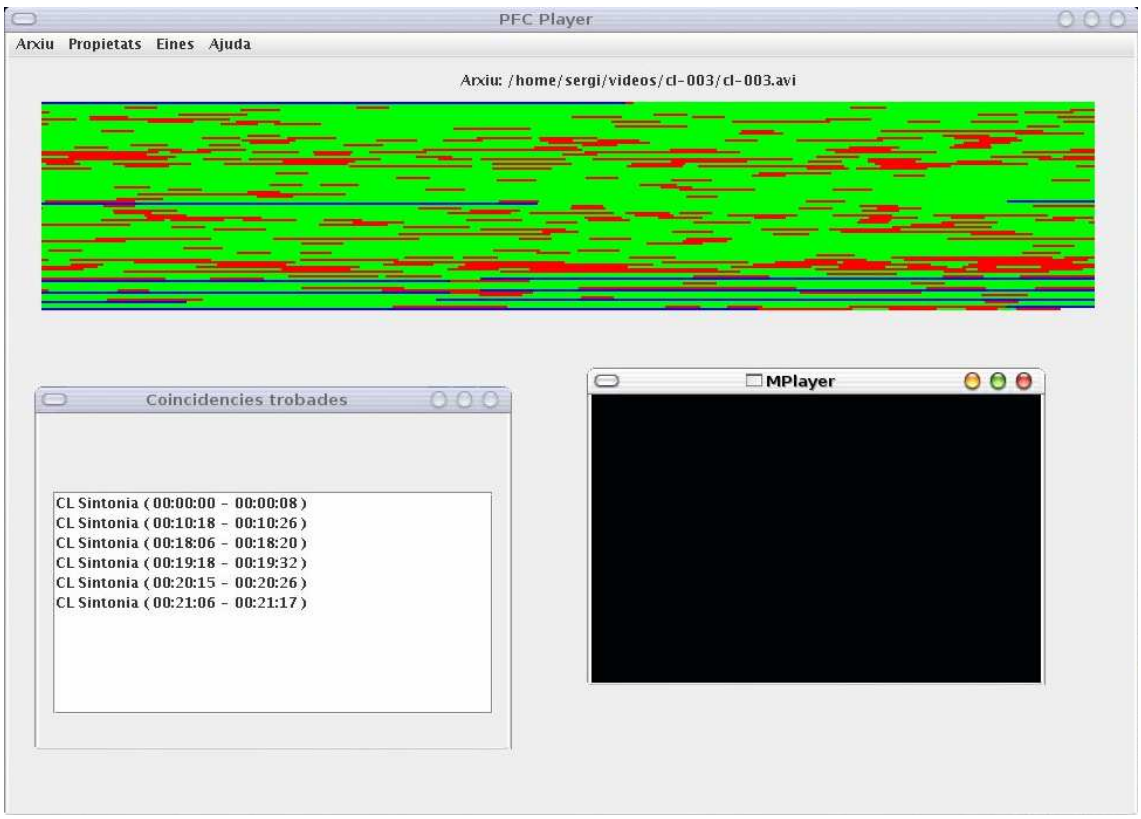
També té la funcionalitat de realitzar consultes al servidor de reconeixement de sons, al qual li realitza consultes amb una *Sliding window* amb solapament. Des de les opcions del programa es pot indicar la mida de la finestra i el solapament amb la anterior, en segons. Quan més petita sigui la finestra i més gran sigui el solapament, més peticions de consulta haurem de realitzar al servidor de reconeixement de sons.

La informació extreta del servidor de reconeixement de sons és mostrada per pantalla en forma de text, on indica el nom del so trobat, el seu inici i el seu final. També és mostrada de manera gràfica a través de la imatge anteriorment creada, on marcarà amb un altre color les parts on ha reconegut so.

La llegenda de colors utilitzada ha estat la següent:

- ● : No s'ha detectat parla.
- ● : S'ha detectat parla.
- ● : S'ha detectat un so a través del servidor *Music Retrieval*.

Una vegada hem obtingut la imatge que representa el nostre arxiu de vídeo, podem escoltar, o bé visualitzar, el moment que desitgem *clickant* amb el ratolí a sobre del punt que ens interessi.



Captura de pantalla de l'aplicació PFC Player. A la part superior tenim la representació gràfica del que representa el so de l'arxiu. Les parts verdes corresponen a fragments de so on existeix parla. La part vermella correspon a les parts on no existeix parla. Les parts blaves son les zones on el detector de sons ha detectat un so que té emmagatzemat a la seva base de dades. A la part inferior esquerra ens mostra la llista de sons detectats. Observem que ha detectat sis vegades la sintonia del telenotícies del Canal Latino. Finalment, a la part inferior dreta trobem la pantalla del vídeo. Aquesta pantalla ens mostrarà 10 segons del punt exacte on *clickem* amb el ratolí a la imatge de la part superior.

Aquesta aplicació s'ha realitzat amb Java i utilitzant la API de *Swing* per a crear l'entorn gràfic. Requereix disposar del reproductor *opensource Mplayer* [18].

S'ha optat per a fer servir un reproductor ja creat perquè la API de Java que s'encarrega de llegir arxius de vídeo, anomenada *Java Media Framework*, no s'actualitza des del 2004 i suporta molts pocs formats de vídeo. Així docs, per estalviar-nos els problemes d'haver de comprimir el vídeo en un format totalment desactualitzat hem fet servir un reproductor *opensource*, que a més a més té la possibilitat d'executar-lo des de línia de comandes, indicant-li el punt inicial i la duració en *frames* que desitgem visualitzar / escoltar.

S'ha inclòs, com annex a aquesta memòria, el diagrama de classes estàtic de UML. Destacar que s'ha creat un package que és capaç de llegir les *speech features* en format XML i convertir-les en un objecte d'una classe de Java.

Els principal problema que ha plantejat aquesta aplicació ha estat la gran mida dels arxius multimèdia, els quals era impossible carregar-los en memòria degut a que superàvem la mida del *Heap* de Java. El que s'ha fet és carregar-los de minut en minut i aplicar la finestra lliscant a cada minut per separat.

A continuació expliquem els diferents resultats obtinguts amb aquesta aplicació:

3.3.2 Resultats

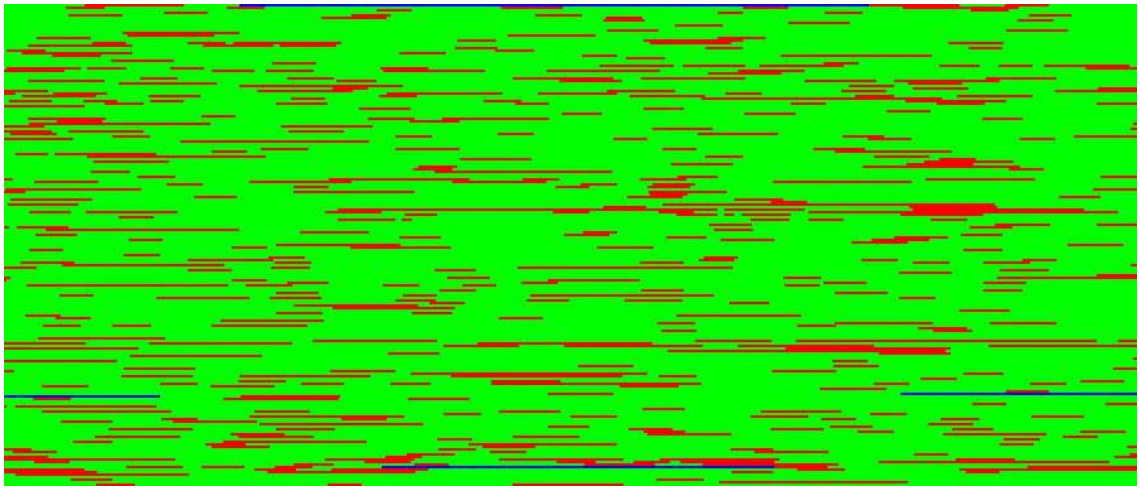
Hem realitzat una sèrie d'experiments amb aquesta aplicació per tal de comprovar la fiabilitat de la detecció de parla de les *Speech Features*, i a la vegada comprovar la robustesa del mètode de detecció de sons utilitzat. A continuació detallem els experiments realitzats i els seus resultats.

3.3.2.1 Anàlisi d'un telenotícies.

Degut a que aquest ha estat el format de programa escollit per a fer els nostres experiments següents, resultava obvi que feia falta testejar totes dues eines (les *Speech Features* i el *Music Retrieval*) amb aquest tipus de programa. Les proves han estat prou satisfactòries.

Per una banda, el reconeixedor de sons ha complert satisfactòriament la seva tasca. Efectivament, aquesta aplicació ha estat capaç de reconèixer els diferents fragments del vídeo on existia el so enregistrat a la base de dades. Donat que és un mètode que pot funcionar amb so que estigui gravat amb distorsió i mala qualitat, al posar-ho en pràctica amb el vídeo enregistrat, que no té cap mena de soroll ni distorsió addicional de l'original, ha donat els resultats esperats.

Per altra banda, les *Speech Features* han funcionat de manera més que acceptable. No es pot negar que en casos en que hi hagi un elevat grau de sons addicionals o de distorsions, comet certs errors alhora de detectar la parla. Aquesta és la sortida per un telenotícies:



Imatge de sortida d'un telenotícies d'Antena 3. La sortida és la típica per aquest tipus de programes. Existeix una gran majoria de fragments amb parla, i els silencis dels presentadors son escassos. Aquests silencis acostumen a tenir una duració molt breu. Al inici del programa és habitual detectar sons emmagatzemats a la base de dades corresponents a la sintonia del telenotícies, així com als intermedis i al final del telenotícies.

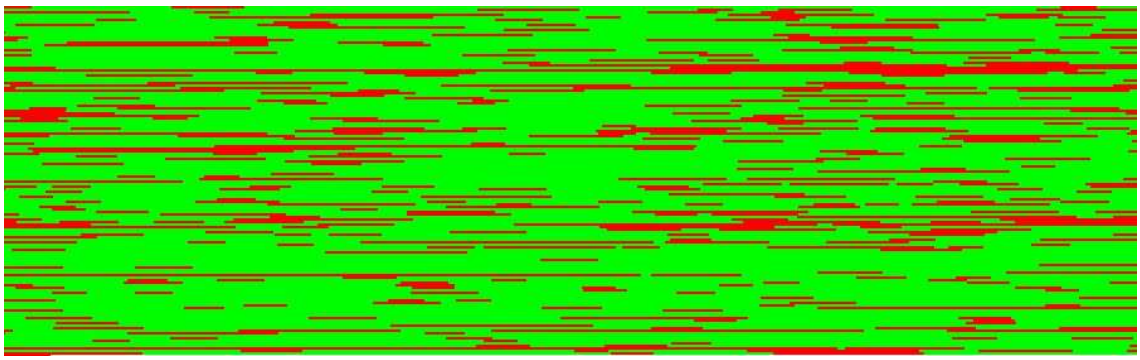
3.3.2.2 Anàlisi d'un documental

Un curiós experiment que hem realitzat amb aquesta aplicació consisteix en analitzar un programa documental.

La idea que volem demostrar, o més ben dit, comprovar, és que existeix un fort contrast entre el nombre d'estats amb parla (i la seva longitud) d'un telenotícies i un documental. Evidentment, l'objectiu d'un programa de notícies consisteix en informar a l'espectador. Les imatges no son suficients per a realitzar-ho, així que constantment existeix una veu, ja sigui la del presentador, o bé una veu en *off* que està informant-nos de les imatges que estem visualitzant.

Contrastant amb aquest estil de programa, existeixen els documentals, i més concretament, els documentals d'animals. En aquest tipus de programes, generalment el que es fa és combinar fragments en els que el presentador ens explica algun fet, i fragments on no hi ha cap veu que ens informi sobre les imatges, bé perquè parlen per elles mateixes, o bé perquè el narrador ja ens ho ha explicat i esta esperant a que acabi d'ocórrer l'acció.

Això dona lloc a trobar pauses molt més llargues que les d'un telenotícies. De passada es pot comprovar la resistència al soroll en entorns on existeix un cert nivell de soroll no massa elevat. Ens referim als sorolls de la natura. La resposta ha estat la esperada:



Sortida del PFC Player per un documental de mamífers.

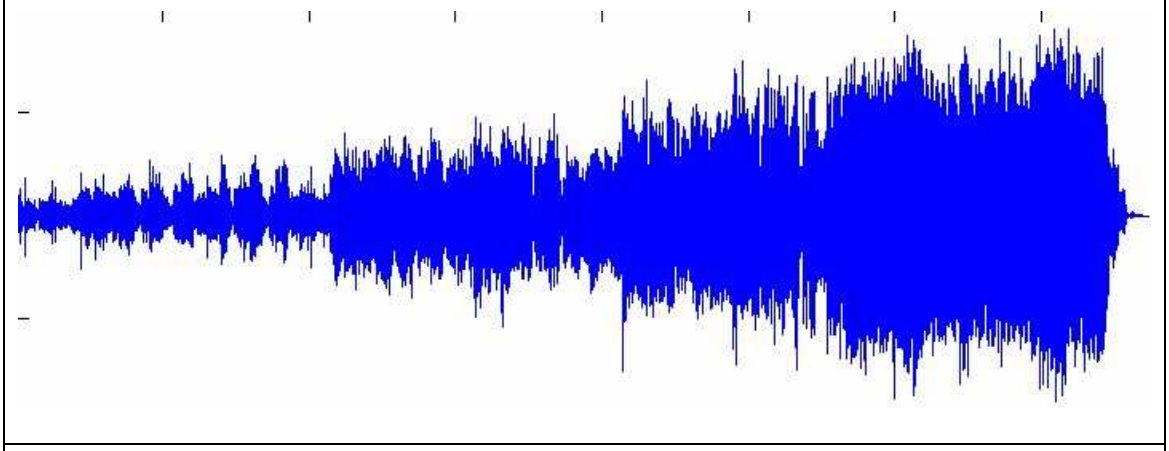
Trobem grans franges horitzontals vermelles que ens indiquen un llarg període de temps sense parla. Resulta curiós el contrast amb un telenotícies, on no existeixen aquestes llargues pauses.

El nostre estudi s'ha centrat en els programes de notícies, però aquest experiment ens pot donar una pista sobre com realitzar un detector de tipus de programes a partir de la freqüència de la parla i de les pauses de la veu. Inclús, si volguéssim enfocar el problema d'una manera més original i curiosa es podria fer analitzant la imatge resultant amb tècniques de visió per computador.

3.3.2.3 Reconeixement de parla en les cançons

Hem testejat aquesta aplicació amb l'entorn més complicat que es pot trobar per analitzar un arxiu de so, i aquest és, sens dubte, un arxiu musical. Les proves realitzades no han estat massa òptimes, i això ens fa notar un fet inqüestionable. El sistema de detecció de les *Speech Features* no és resistent al soroll elevat. En un arxiu musical la veu no està gairebé mai sola, i això dificulta l'anàlisi de l'àudio. La *Toolbox* creada pel *MIT Media Lab* inclou una part de reducció del soroll, però està clar que encara falta molt per avançar en aquest sentit.

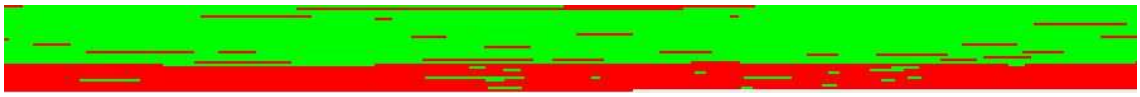
La cançó testejada [19] comença amb una introducció calmada a base de guitarres acústiques i flautes, gradualment va cedint-li el pas a una secció intermèdia elèctrica lenta, fins culminar amb una última part més accelerada i elèctrica. En la seva resposta en amplitud es pot veure clarament aquest progrés:



Amplitud del senyal de [19]

Aquesta cançó ha passat pel procés d'obtenció dels arxius XML, i els resultats han resultat força clarificadors. Podem afirmar clarament que la *Toolbox* de les *Speech Features* no suporta les cançons musicals. No és capaç de detectar la parla amb entorns on hi ha un elevat índex de soroll.

La resposta que hem obtingut amb la nostra aplicació és el següent:



Sortida del PFC Player per [19].

La sortida resulta força curiosa. La introducció a base de guitarres acústiques i flautes la detecta com a parla, quan en realitat no és així, i continua detectant parla fins que la cançó es torna més elèctrica. Evidentment, la última part, on la cançó evoluciona fins a culminar en una obra de rock dur, encara detecta amb menys encert la parla del cantant.

Evidentment, la aplicació no serveix per a detectar la parla en obres musicals, però podria servir per a realitzar un mapa de la cançó que senyali les parts més relaxades i les més dures i agressives. Podria resultar una alternativa bastant més simplista al *Emotiongram* [16].

3.4 Segmentació de telenotícies

La part important del nostre projecte s'ha centrat en analitzar els telenotícies de quatre emissores diferents i, a partir de característiques de l'àudio i del vídeo dels programes, ser capaços de distingir-los automàticament amb mètodes d'aprenentatge supervisat.

Les emissores escollides han estat les següents: TV3, Antena 3, Canal Latino i TV Sant Cugat. Varem decidir escollir-ne dues totalment professionals i dues més d'un format més senzill.

Aquest projecte s'ha centrat en l'anàlisi del so, mentre que el meu company Jordi Hernández ha focalitzat els seus esforços en la extracció de dades de la imatge.

3.4.1 Característiques extretes

Referent a la part del so, hem extret 6 característiques. Son les següents:

- To.
- Sonoritat.
- Mida de les frases.
- Temps transcorregut entre frases.
- Mida de les síl·labes.
- Temps transcorregut entre síl·labes

El to i la sonoritat s'han extret directament de les *Speech Features* calculades prèviament. En canvi, la resta s'ha calculat a partir dels estats amb parla / no parla i dels estats amb sons vocals / no vocals. S'han calculat de la següent manera:

- $Mida\_de\_les\_frases = \frac{num\_Estats\_parlant}{num\_frases}$
- $Temps\_entre\_frases = \frac{temps\_total - temps\_parlant}{num\_frases}$
- $Mida\_de\_les\_sil{\cdot}labes = \frac{temps\_estats\_vocals}{num\_silabes}$
- $Temps\_entre\_sil{\cdot}labes = \frac{temps\_total - temps\_sons\_vocals}{num\_sil{\cdot}labes}$

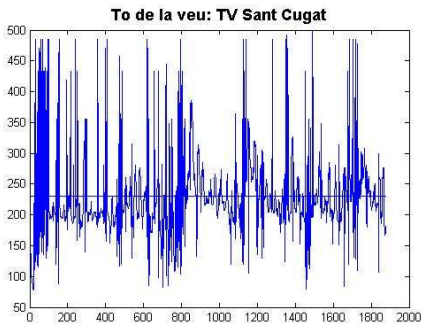
Com es pot observar, resulta prou senzill calcular aquests paràmetres a partir dels diferents estats en que les *Speech Features* divideixen el telenotícies.

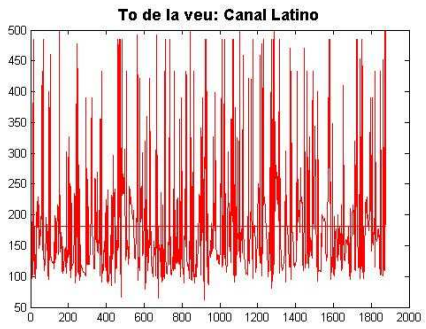
3.4.1.1 To

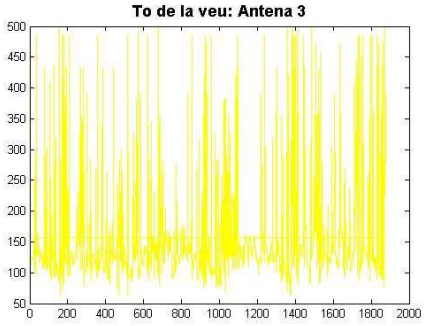
Aquesta característica és extreta de la Toolbox de les Speech features.

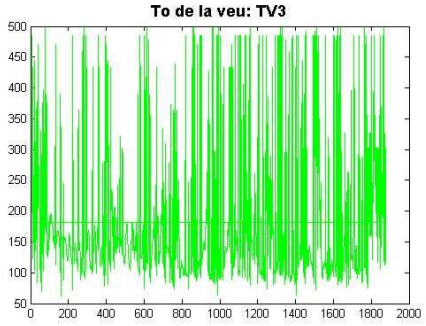
Cada presentador de telenotícies té el seu to de veu. Resulta un fet diferencial força bo, però tampoc és definitiu, ja que la mitjana del to de certs telenotícies és força semblant, mentre que pot resultar molt diferent en altres casos.

Aquest és el resultat d'analitzar un fragment de 30 segons de cada presentador:

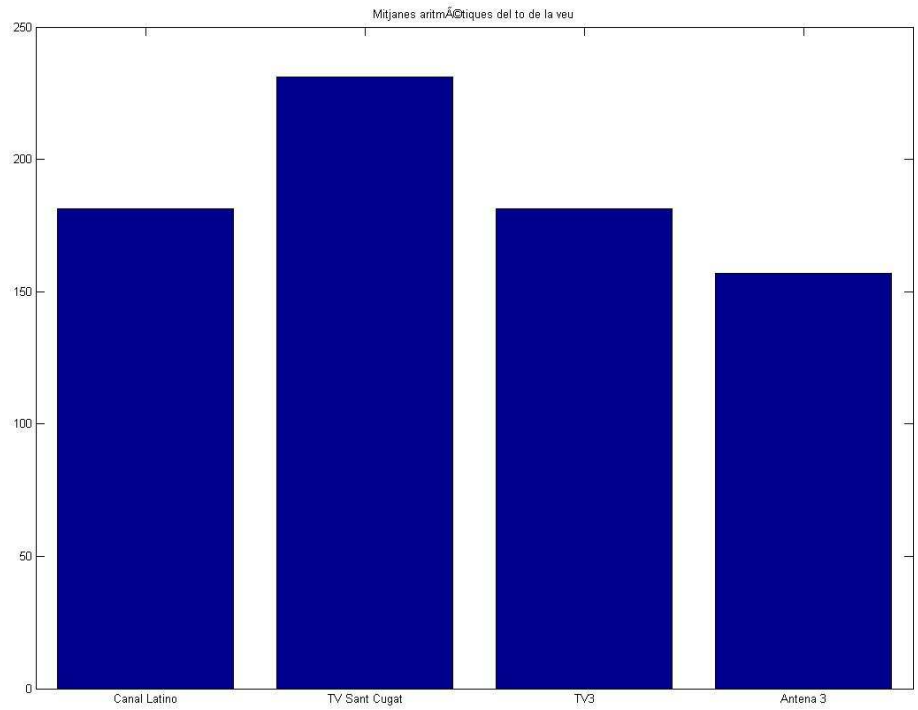








Tal com s'observa a les imatges, el to de la veu dels diversos canals no és regular, però en canvi es pot aplicar la mitjana aritmètica de totes les mostres. Aquesta mitjana aritmètica del fragment escollit és un tret diferencial que explotarem. La línia recta horitzontal que apareix en cada gràfica assenyala la mitjana aritmètica de totes les mostres. Mentre que TV Sant Cugat té un to molt diferenciat de la resta, TV3 i Canal Latino son pràcticament idèntics.

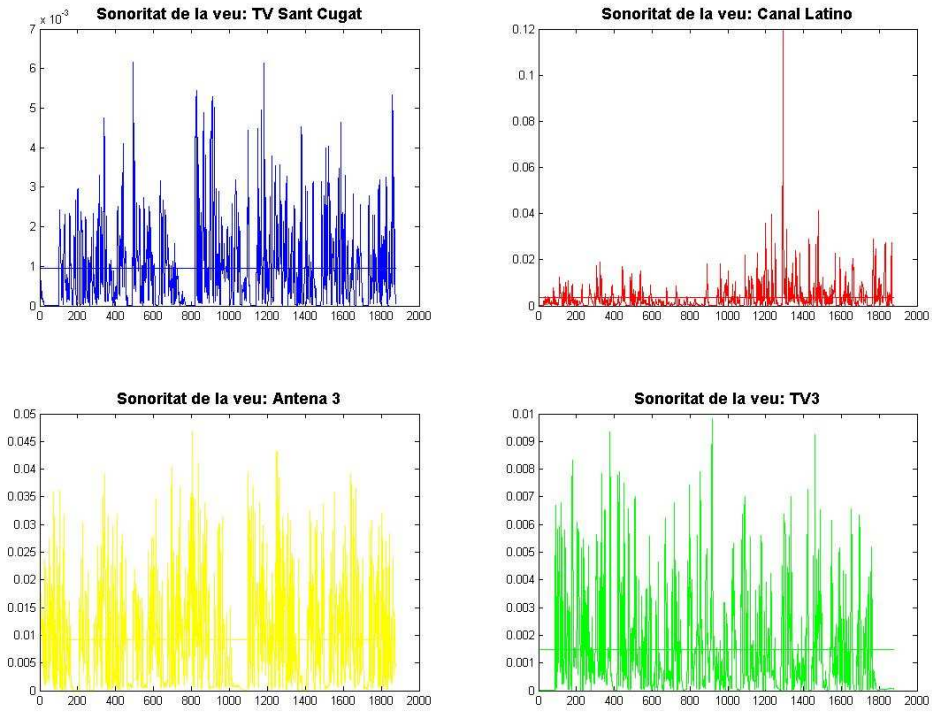


Aquesta gràfica mostra la mitjana aritmètica del to de la veu de cada presentador. És evident que és un tret distintiu, però degut a la mínima diferència entre certs canals no és un tret diferencial definitiu.

3.4.1.2 Sonoritat

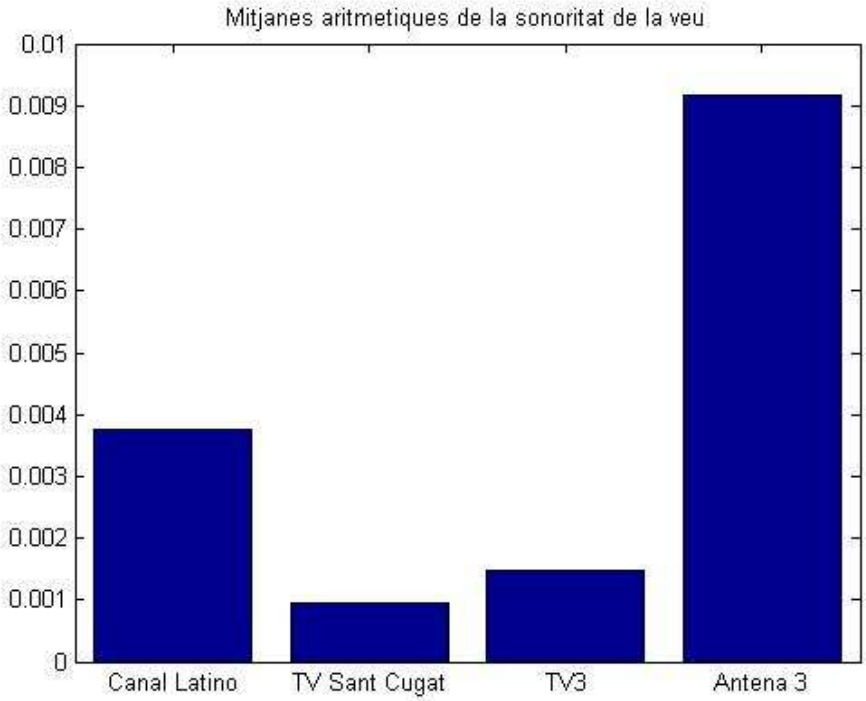
Aquesta característica resulta bàsica a l'hora de realitzar la segmentació. Al realitzar proves de volum entre els 4 telenotícies ens trobem en que la diferència entre ells és prou notable. Com en el cas anterior, és extreta directament de les *Speech Features*.

Aquest és el resultat d'analitzar un fragment de 30 segons de cada presentador:



Com passava en el to de la veu, en un fragment de telenotícies ens trobem forts canvis de sonoritat, però la mitjana aritmètica (marcada amb una línia recta horitzontal) és prou significativa.

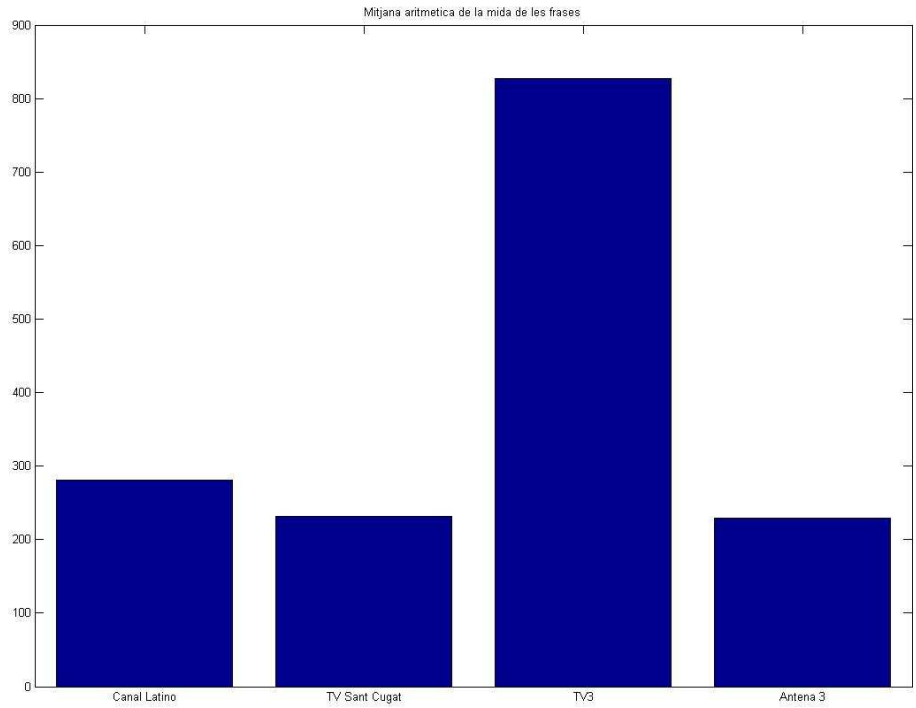
El diagrama de barres amb la mitjana aritmètica del to de la veu accentua de manera notable la diferència de sonoritat entre els diferents telenotícies escollits.



Com es pot observar a la gràfica existeix una forta diferència entre Antena 3 i la resta.

3.4.1.3 Mida de les frases

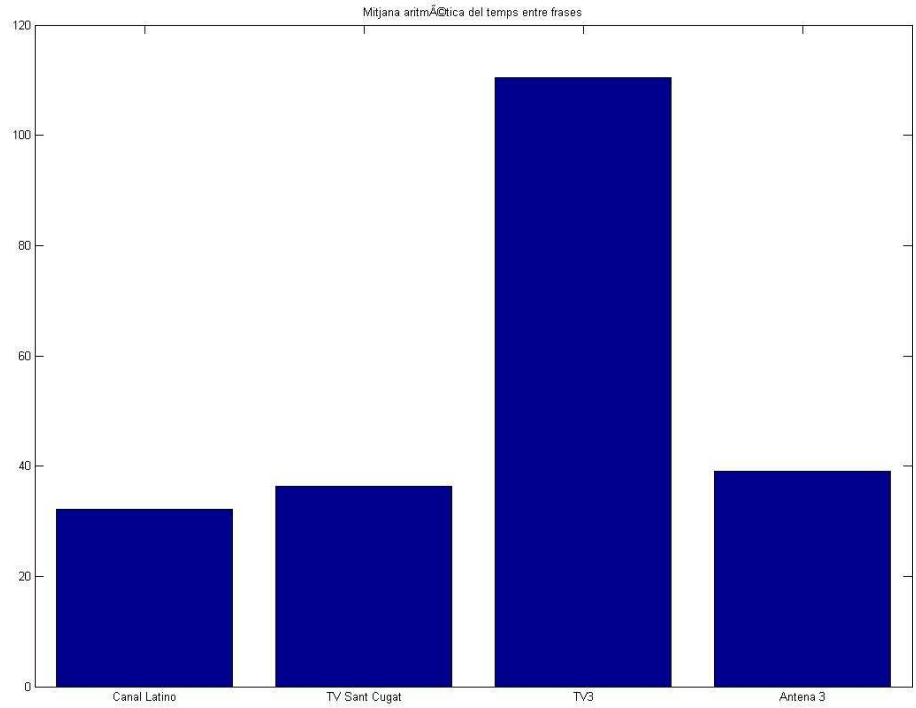
Aquesta característica resulta més útil si el fragment a analitzar és un fragment d'un llarg període de temps, ja que si analitzem fragments curts la mida de les frases acabarà convergint amb la mida del fragment. Hem realitzat la prova amb 30 segons de cada telenotícies i els resultats són força sorprenents.



El resultats de la prova han destacat la llarga longitud de les frases en el telenotícies de TV3 en comparació amb la resta. La resta de cadenes tenen una longitud força semblant per una longitud de fragment de 30 segons.

3.4.1.4 Temps entre frases

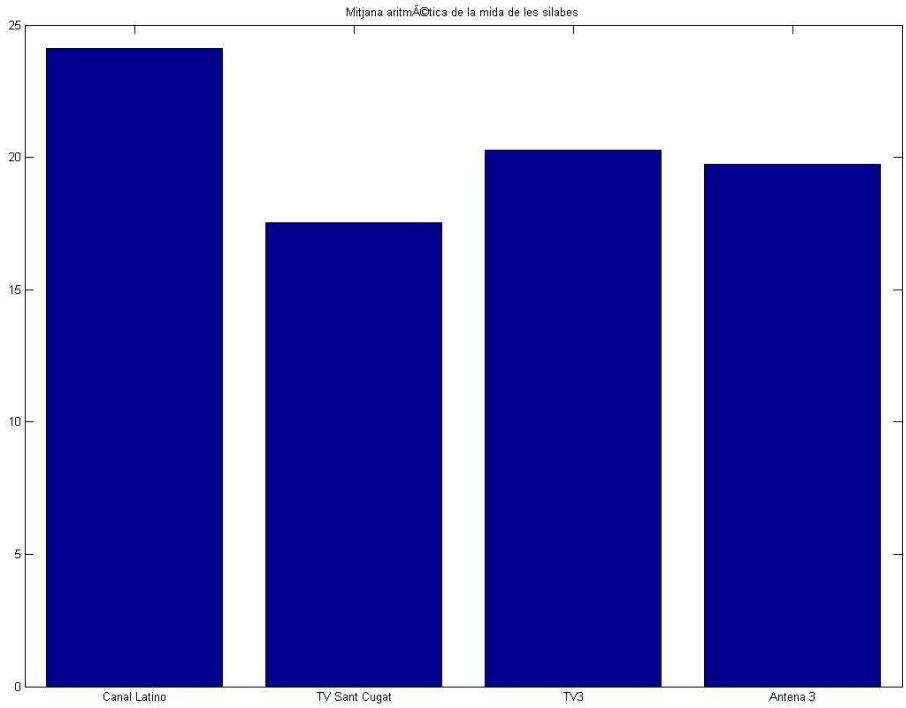
Aquesta mesura també ha destacat certes diferències entre les diferents emissores. El següent diagrama de barres mostra aquests resultats.



Les pauses entre els diferents telenotícies resulten molt semblants, excepte en el cas de TV3. En aquest darrer cas, es pot observar que la duració d'aquestes pauses ha resultat de més del doble.

3.4.1.5 Mida de les síl·labes

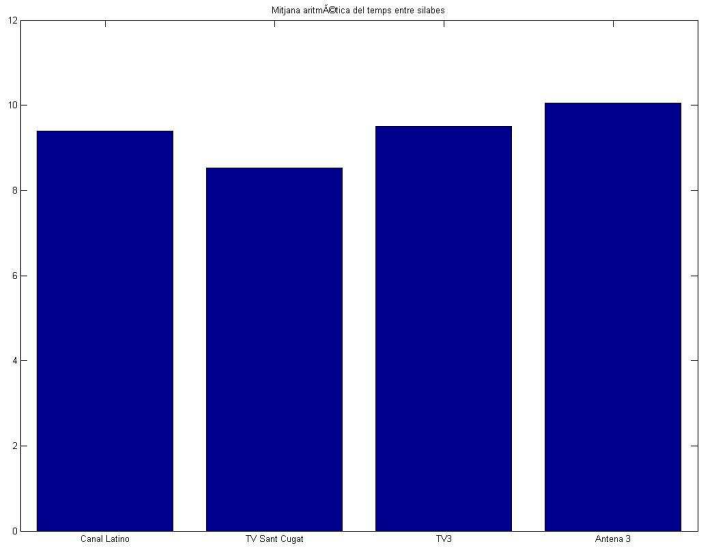
Aquesta característica no mostra una gran diferència entre els diferents telenotícies, ja que estem treballant sobre dues llengües que son molt semblants. Aquesta característica podria destacar si es comparés amb altres idiomes on la estructura i la articulació de les paraules sigui més diferent a la catalana i castellana. Aquests són els resultats:



Com s'observa en el gràfic, les diferències no són excessivament notables.

3.4.1.6 Temps entre síl·labes

Com ha passat en la anterior característica, aquesta no és una característica fonamental a l'hora de diferenciar els diferents telenotícies. Aquests han estat els resultats d'analitzar aquesta característica amb 30 segons de cada telenotícies.



Igual que en el cas anterior, aquesta característica no presenta diferències notables entre els diferents telenotícies.

3.4.2 Classificador de telenotícies

A partir de les característiques detallades anteriorment, hem construït una aplicació que reconeix i classifica els diferents telenotícies. Hem utilitzat 13 classificadors diferents, dels quals hem explicat les seves principals característiques en un document annex a aquesta memòria. Tots aquests classificadors pertanyen a la *Toolbox PRTools4* [23]. Són els següents:

1. Bayesià lineal
2. Bayesià quadràtic
3. *Parzen*
4. Xarxa neural amb *Backpropagation* de 3 unitats ocultes.
5. Lineal amb expansió *Karhunen–Loève* de la matriu de convolució
6. Lineal amb expansió *PCA*
7. Lineal logístic
8. Lineal *Least Squared Error*
9. Mixtura de gaussianes

10. K veí més proper
11. Densitat *Parzen*
12. Xarxa neural amb *Levenberg-Marquardt*
13. Xarxa neural amb *radial basis*

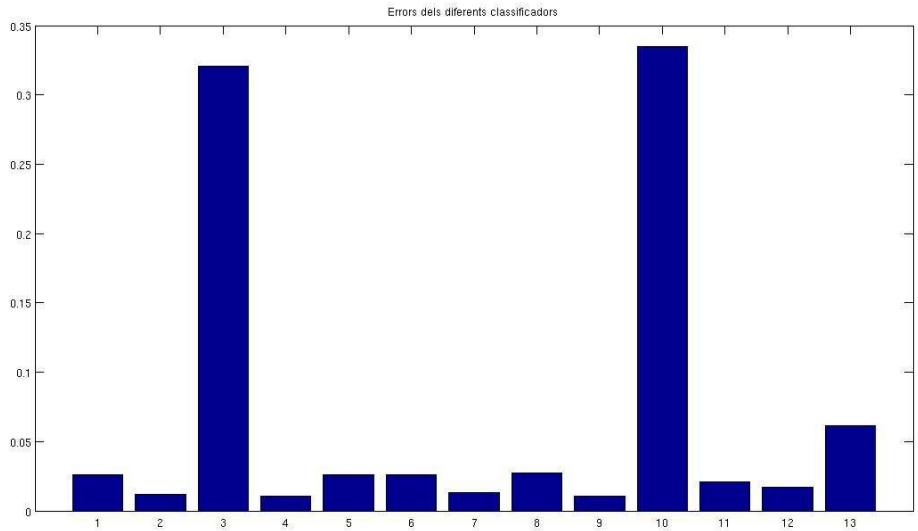
Les proves s'han realitzat amb un nombre de mostres d'aprenentatge de 250 mostres per a cada emissora, a més d'un conjunt de test de 1372 mostres. Cadascuna d'aquestes mostres s'ha extret de 5 segons de vídeo amb un solapament entre ells d'un segon. A continuació mostrem els resultats obtinguts.

3.4.2.1 Resultats

Cal dividir les proves realitzades en dos grups. Inicialment s'ha provat el correcte funcionament dels classificadors únicament amb la part del so. Finalment hem realitzat les proves unint la feina realitzada en l'anàlisi de la imatge amb la part del so, i d'aquesta manera crear un sistema complet. Els resultats han estat els següents:

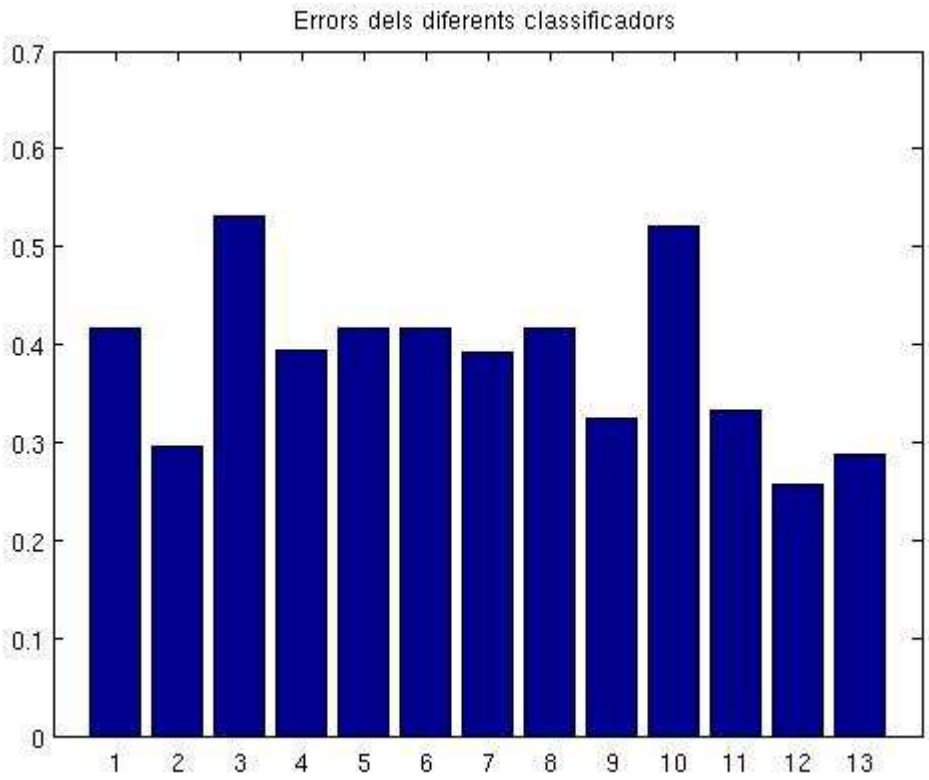
3.4.2.1.1 Classificador d'àudio

En aquest cas, s'han donat uns resultats una mica contradictoris. Si prescindim del Canal Latino, els resultats han estat els següents:



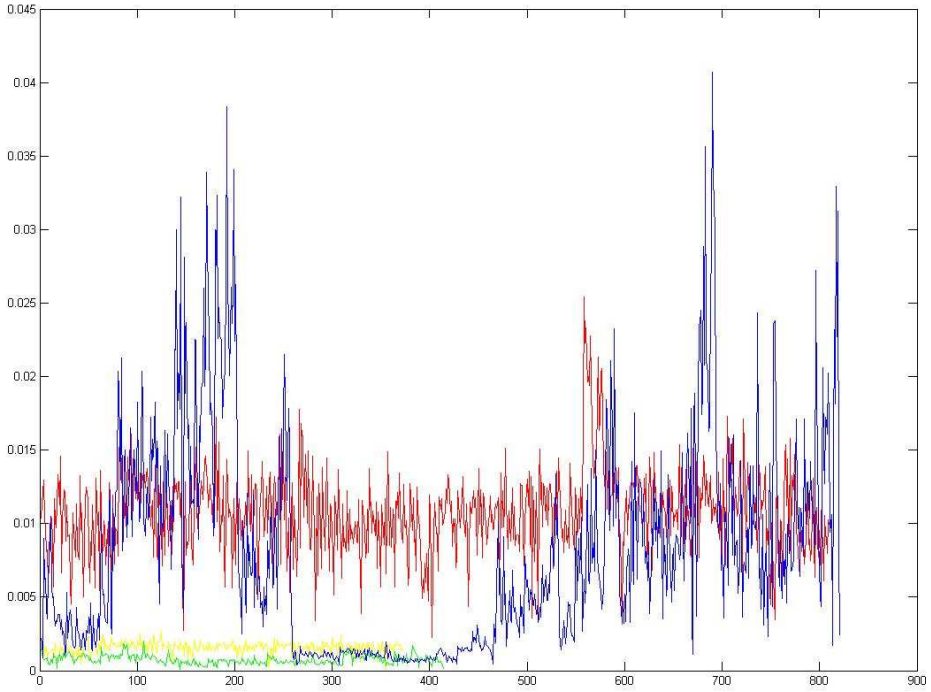
Exceptuant els resultats del classificador Parzen i el K veí més proper, els resultats son força bons, ja que obtenim un error que oscil·la entre el 2% i el 3%

En canvi, si realitzem la mateixa prova amb el Canal Latino, els resultats són força diferents:



A l'incloure el Canal Latino, l'error ha augmentat notablement, oscil·lant entre el 25% i el 50%

La causa d'aquest motiu és la mala qualitat del senyal en la recepció d'aquesta emissora, on les propietats del so varien en cada emissió. La característica que es veu principalment afectada és la sonoritat.



Sonoritat de les cadenes Antena 3 (vermell), TV3 (groc), TV Sant Cugat (Verd) i Canal Latino (Blau). Com s'observa en el gràfic, la sonoritat de les diferents cadenes és força constant, i es pot traçar una mitjana que és realment significativa. Excepte en el cas de Canal Latino, on la seva sonoritat varia de forma desmesurada degut a que cada emissió és diferent de la resta. Això fa que aquesta característica sigui capaç de ser un tret distintiu de totes les emissores excepte Canal Latino.

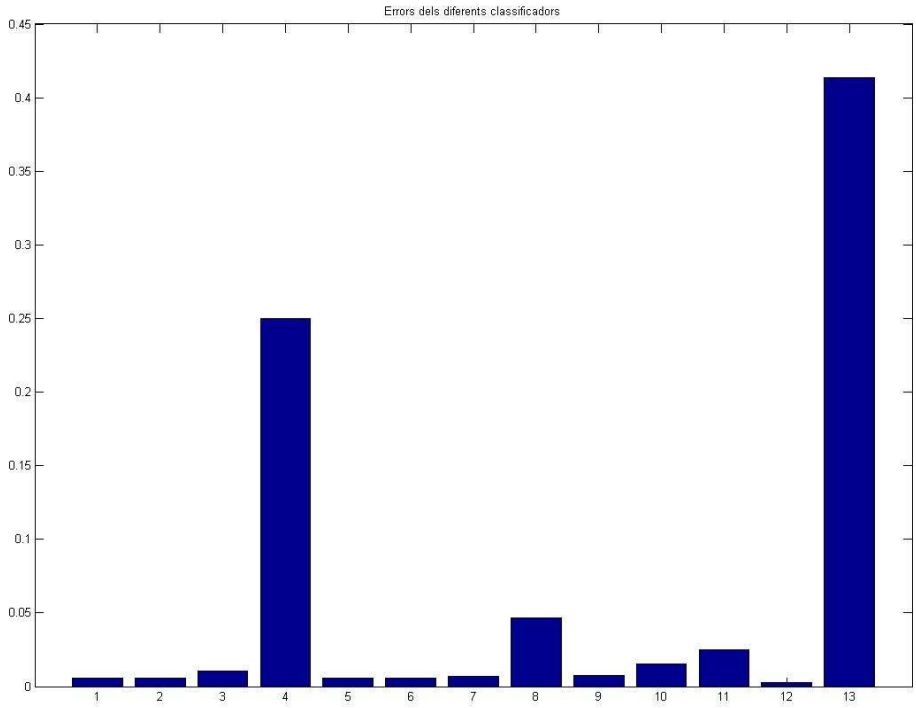
El motiu d'aquest problema és que les emissions del Canal Latino són en format analògic i amb poca qualitat. Així doncs, si el classificador combinat de vídeo i àudio és capaç de distingir Canal Latino de la resta, el classificador funcionarà correctament.

3.4.2.1.2 Classificador d'àudio i vídeo

La última part d'aquest projecte ha consistit en la unificació del projecte realitzat per Jordi Hernández sobre les característiques de vídeo i la part d'anàlisi del so explicada en aquesta memòria.

S'ha realitzat un classificador que conté les 7 característiques referents al so, a més de 8 característiques referents a la imatge.

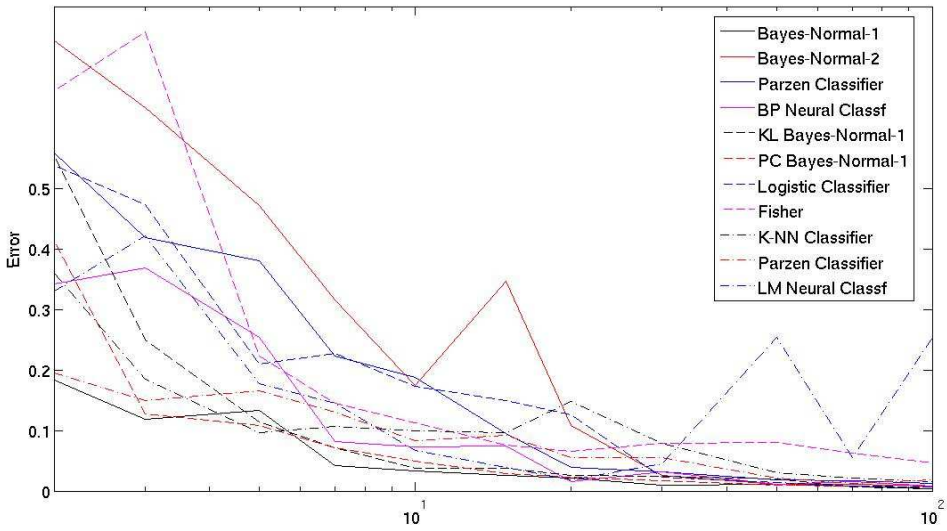
Els resultats obtinguts d'aquest classificador conjunt han resultat un èxit. A continuació mostrem l'error obtingut en cada classificador:



Exceptuant el cas del classificador *Back Propagation* i *Radial Basis*, la resta funcionen molt bé, i en la majoria dels casos l'error no arriba al 1%. Així doncs, és possible la realització d'un classificador que distingeixi clarament entre els diferents telenotícies.

Amb les proves realitzades, la xarxa neural *Levenberg-Marquardt* és el classificador que obté uns millors resultats, on l'error oscil·la entre el 0,2% i el 0,3%.

Aquesta és l'evolució de l'error en funció del nombre de mostres d'entrenament:



Tots els classificadors, exceptuant els de xarxes neurals *Back Propagation* i *Radial Basis*, que no acaben de funcionar correctament, s'estabilitzen cap al voltant de les 150 mostres, i al voltant de 200 l'error és pràcticament nul.

3.4.2.1.3 Ranking de Característiques

Les característiques del so més importants per al classificador han estat, de major a menor importància, les següents:

1. Volum
2. To
3. Mida de les síl·labes
4. Temps transcorregut entre frases
5. Temps transcorregut entre síl·labes
6. Mida de les frases

Aquesta classificació és del tot lògica. Al treballar amb fragments de cinc segons, és normal que la mida de les frases tingui poca importància, ja que la mida màxima serà de cinc segons. Per altra banda, la mida de les síl·labes guanya importància pel mateix motiu anterior.

Així doncs, podem afirmar que les característiques més destacables han estat les més bàsiques, en comptes de les referents a l'anàlisi de les frases. Això canviaria si

treballéssim amb fragments de 30 segons de programa, on les diferències entre mides de frase es farien més evidents.

CONCLUSIONS

En aquest capítol es relacionaran els objectius aconseguits i els no aconseguits, així com ara les principals aportacions que creiem que tenen més valor, i les possibles línies de continuació d'aquest treball, indicant que podríem millorar i/o aprofundir.

4 Conclusions

4.1 Objectius assolits

Després de realitzar el nostre projecte hem assolit el nostre principal objectiu. Ens trobem en condicions d'afirmar que és possible la indexació automàtica de continguts televisius a partir de la extracció de característiques referents al so i a la imatge.

Si bé és cert que el so, a diferència de la imatge, és especialment sensible a les distorsions i al soroll, aquest és un problema mínim degut al canvi que s'està produint en el món de la televisió, on totes les emissions passaran al format digital eliminant aquest tipus d'interferències. Això facilitarà la homogeneïtat en les dades extretes de les diferents mostres.

També cal destacar que, en el cas que alguna d'aquestes característiques pateixi variacions degut a les distorsions del senyal, com ha passat amb la sonoritat d'alguna cadena, no afecta al resultat òptim del classificador. Això ens permet demostrar que és un sistema robust d'indexació de continguts televisius.

Aquesta robustesa ha estat possible gràcies a que, en el cas que el so o la imatge sofreixi distorsions, serà possible avaluar correctament els resultats degut a la unió amb l'altre conjunt de característiques.

Malauradament, si la emissió és de mala qualitat, el conjunt de característiques que primer sofreix les conseqüències és el so, ja que estem treballant amb valors molt concrets de característiques del senyal, com ara la sonoritat i el to. En una imatge, al tractar-se d'un anàlisi en un nivell molt més superior, com ara la mida de les cares detectades, no afecten tant les distorsions i interferències del senyal. Aquest és un problema que desapareixerà, ja que amb la televisió digital, o bé es sent el so amb qualitat òptima, o bé no existeix so.

4.2 Objectius no assolits

L'objectiu que no ha estat possible aconseguir és la possibilitat de processar i extreure les dades dels vídeos a temps real. Degut a l'elevat nombre de càlculs necessaris per a la extracció de les *Speech Features*, així com la gran dimensió de la mida de les dades, fa impossible l'anàlisi d'aquestes dades i la extracció de resultats a la mateixa velocitat que es visualitza el vídeo.

De fet, aquesta ha estat la nostra principal dificultat, ja que el processament i extracció de dades d'aquests arxius de vídeo és un procés lent i que requereix gran quantitat de memòria on emmagatzemar aquestes dades.

4.3 Principals aportacions

Les aportacions d'aquest projecte que tenen més valor, segons el nostre criteri, han estat les següents.

Per una part, s'ha creat una interfície que permet la extracció de les característiques referents al so, i aquestes característiques són exportades en format XML per al seu posterior anàlisi en qualsevol llenguatge de programació que suporti XML, que són la gran majoria.

Per altra banda, s'ha creat un conjunt de classes en Java que permeten la lectura d'aquests XML's per al seu ús dins d'aquest llenguatge de programació, tal com hem fet en l'aplicació PFC Player.

I finalment, el que creiem que és l'aportació més valuosa d'aquest projecte, ha estat la extracció de dades quantificables i diferenciadores dels diferents programes de televisió per a la seva posterior indexació. Aquestes dades han consistit en el to, la sonoritat, la mida de les frases, l'espai entre frase i frase, la mida de les síl·labes i l'espai entre síl·labes, a part de les característiques referents a la imatge.

La posterior utilització d'aquestes dades en un classificador ha demostrat la robustesa d'aquest sistema, ja que hem patit les conseqüències de soroll i distorsió en el so en un dels programes escollits. Tot i això, l'error dels nostres classificadors no arribaven al 1%.

En el nostre projecte ens hem centrat en l'estudi dels telenotícies, però creiem que son dades suficientment generals com per a ser utilitzades per la indexació de qualsevol tipus de programa. És a dir, l'ús d'aquestes dades no està limitat a l'àmbit dels telenotícies.

4.4 Línies de continuació

En aquest projecte ens hem centrat en la extracció de característiques del so referents a la parla, com poden ser la mida de les frases, o bé la longitud de les síl·labes. En canvi, hi han altres aspectes del so que no han estat tractats en aquest projecte.

Una possible continuació d'aquest projecte es podria centrar en l'anàlisi dels sons no referents a la parla, és a dir, seleccionar els fragments on no es detecta activitat de parla mitjançant les *Speech Features*, i a partir d'aquí, extreure característiques referents al so ambiental. La indexació de continguts televisius es podria realitzar mitjançant un classificador de les característiques combinades de la parla i del so ambiental.

Un altre aspecte que es podria tenir en compte seria l'anàlisi de les emocions a partir de les proves visuals i el so. Aquest és un tema complex degut a que les emocions en l'ésser humà son reaccions complexes, i resultaria difícil establir una lògica en elles. Podríem arribar a pensar que les emocions són totalment prescindibles, però és evident que existeixen. Segurament seria possible la extracció de característiques a partir de les emocions que expressen els personatges dels programes televisius. Inclús seria possible la extracció de característiques a partir de les emocions generades sobre el televident. Evidentment, és un tema prou complex com per a ocupar varis projectes per ell sol.

5 Referències i Bibliografia

[1] Yan Ke, Derek Hoiem, Rahul Sukthankar. *Computer Vision for Music Identification*.

[2] Michael Fink. *Social – and Interactive – Television. Applications Based on Real-Time Ambient-Audio Identification*.

[3] William T.Stoltzman. *Toward a Social Signaling Framework: Activity and Emphasis in Speech.*

[4] Michele Covell, Shumeet Baluja, Michael Fink. *Detecting Ads in Video Streams Using Acoustic and Visual Cues.*

[5] Mathworks Matlab. *Specgram help.*

[6] Ignasi Serra i Pujol, Ramon Vilanova i Arbós. *Tractament del senyal.*

[7] http://www.ais.fraunhofer.de/~surmann

[8] http://googleresearch.blogspot.com/

[9] Michele Covell, Shumeet Baluja. *Known-Audio Detection Using Waveprint: Spectrogram Fingerprinting By Wavelet Hashing*.

[10] Shumeet Baluja, Michele Covell. *Audio Fingerprinting: Combining Computer Vision & Data Stream Processing.*

[11] Eugene Weinstein, Pedro Moreno. *Music Identification With Weighted Finite-State Transducers.*

[12] Shumeet Baluja, Michele Covell. *Learning "Forgiving" Hash Functions: Algorithms and Large Scale Tests.*

[13] http://www.hitsongscience.com/

[14] Enderroc, núm. 116

[15] Craig Anderton. *EQ, 2000; issue 3. The quantification of Emotion*.

[16] Meghen Miles, Merrick Mosst. *Emotiongram, Visualitzing Emotional Content in Audio*. http://www-scf.usc.edu/~ise575/b/projects/mosst_miles/concept.htm

[17] Russell, Norvig. *Inteligencia Artificial. Un enfoque moderno*.

[18] http://www.mplayerhq.hu/

[19] Jimmy Page, Robert Plant. *Stairway to Heaven; Led Zeppelin; IV*.

[20] Sumit Basu. *Conversational Scene Analysis*.

[21] Ronc Caneel. *Social Signaling in Decision Making.*

[22] Van der Heikden, Duin, de Ridder, Tax. *Classification, Parameter Estimation and State Estimation. An Engineering Approach using Matlab*

[23] http://www.prtools.org

[24] http://www.youtube.com

[25] http://www.joost.com

[26] *International Phonetic Association*. http://www2.arts.gla.ac.uk/IPA/fullchart.html

6 Documents Annexes

6.1 L'espectrograma

L'espectrograma és un concepte força senzill, però per altra banda fonamental en aquest projecte, que consisteix en el següent:

S'aplica una senyal d'entrada, normalment so. L'eix horitzontal representa el temps, mentre que l'eix vertical representa la freqüència. Els canvis de color en la imatge consisteixen en els diferents nivells d'intensitat que té la senyal.

Per a construir-lo es pot aplicar el següent algoritme [5] [6] :

Amb una senyal S d'entrada:

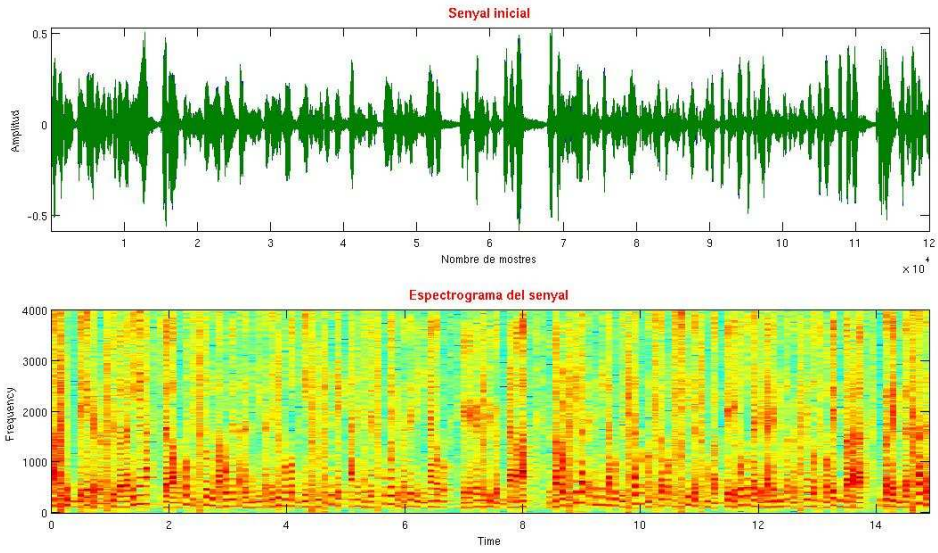
Es parteix la senyal en fragments solapats entre ells, es a dir, és una *Sliding Window* amb solapaments.

Es calcula la transformada de Fourier de temps discret per cada finestra.:

$$X(m,\omega) = \sum_{n=-\infty}^{\infty} x[n]\omega[n-m]e^{-j\omega n}$$

Aplicant el mòdul, cada transformada representarà una columna del nostre Espectrograma:

$$Espectrograma\{x()\} = \left|X(\tau,\omega)\right|^2$$



Exemple d'espectrograma d'una senyal d'àudio inicial.

6.2 Algoritme EM

L'algoritme EM es pot aplicar en moltes situacions en les que faci falta estimar un conjunt de paràmetres σ que descriuen una distribució de probabilitat subjacent, donada únicament per la part observada de les dades completes produïdes per la distribució.

Suposem que *x* representa el conjunt de dades completes, que s'observa a través de *y*. És a dir, existeix una funció *h* tal que $y = h(x)$. Tenim el conjunt $X(Y) = \{x : y = h(x)\}$ i $g(y/\sigma)$, que representa la funció de densitat o de probabilitat de *y*, la qual conté un paràmetre σ desconegut per a nosaltres, el qual es vol estimar. $f(x/\sigma)$ representa la funció de probabilitat del conjunt de dades complet. Les dues probabilitats es relacionen de la següent manera: $g(y/\sigma) = \int_{X(Y)} f(x/\sigma)dx$.

En aquest algoritme es fa la assumpció que és més fàcil treballar amb *f* que amb *g*. La funció *h* no és única, ha de ser determinada de manera que sigui fàcil treballar amb $f(x/\sigma)$.

L'algoritme EM és iteratiu, i estima el valor de σ. L'algoritme és el següent:

Escollir un valor inicial σ_0 tal que $\sigma = \sigma_0$.

Considerar la funció $Q(\sigma/\varphi) = E[\log f(x/\sigma)/y, \varphi] = \int \log f(x/\sigma)k(x/y, \varphi)dx$

On $K(x/y, \varphi) = \dfrac{f(x/\psi)}{g(y/\psi)}$

Sigui $\sigma^{(p)}$ el valor estimat del paràmetre durant la iteració p. Aleshores, el valor estimat durant la iteració p + 1 es calcula de la següent manera:

Pas E (valor Esperat)

Es calcula $Q(\sigma, \sigma^{(p)}$

Pas M (maximització)

Es calcula $\sigma^{(p+1)} = \arg\max_{\sigma} Q(\sigma/\sigma^{(p)})$

El procés iteratiu acaba quan $|\sigma^{(p+1)} - \sigma^{(p)}| < TOL$, on TOL=10^{-6} o 10^{-7} i el valor

estimat serà $\sigma^{*} = \sigma^{(p+1)}$

6.3 Algoritme RANSAC

L'objectiu d'aquest algoritme és trobar un ajustament robust d'un model a un conjunt de dades S, que conté errors que són numèricament distants de la resta de les dades.

L'algoritme és el següent:

1. Seleccionar aleatòriament una mostra de *s* punts del conjunt total *S* i calcular el model amb aquest subconjunt.
2. Determinar el subconjunt de punts S_i que estan dins del llindar de distància al model. El subconjunt S_i es el conjunt **consens** i defineix els valors que son numèricament distants de la resta de les dades de S, anomenats "inliers".
3. Si el conjunt S_i és més gran que un llindar *T*, es torna a estimar el model utilitzant tots els punts de S_i i acabar.
4. Si la mida de S_i és més petita que *T*, seleccionar un nou subconjunt i repetir el pas anterior.
5. Després de *N* intents seleccionar el conjunt S_i amb un consens més gran, tornant a estima el model utilitzant tots els punts el conjunt S_i

6.4 Classificadors

Durant la realització dels nostres experiments hem utilitzat una sèrie de classificadors per a mesurar els nostres resultats. A continuació exposem els trets característics més bàsics sobre el funcionament d'aquest tipus d'aplicacions:

6.4.1 Aprenentatge

És la part bàsica que tenen en comú els diferents tipus de classificadors que existeixen.

La idea bàsica del aprenentatge consisteix en utilitzar les percepcions no només per actuar, sinó també per a millorar la habilitat d'un agent per actuar en el futur [17] .

Existeixen diversos tipus d'aprenentatge:

- Aprenentatge supervisat: Consisteix en aprendre una funció a partir d'exemples de les seves entrades i les seves sortides. No sempre és possible fer aquest tipus d'entrenament ja que hem de disposar de la sortida esperada en funció de la entrada.
- Aprenentatge no supervisat: Consisteix en aprendre a partir de patrons d'entrades pels quals no s'especifiquen els valors de les seves sortides.
- Aprenentatge per reforçament: L'agent ha d'aprendre a partir d'una funció de reforç, que li serveix de realimentació per a valorar la entrada. Un exemple d'aquest tipus d'aprenentatge seria el cas d'un agent que faci la funció de cambrer. Pot servir-li com a reforç la quantitat de propina que el client li ha donat.

La idea de l'aprenentatge consisteix en construir una funció que tingui el comportament observat en les seves dades d'entrada i de sortida. Els mètodes d'aprenentatge es poden entendre com la cerca d'un espai d'hipòtesis per a trobar la funció adequada partint d'una assumpció molt bàsica respecte a la funció.

6.4.2 Classificador Bayesià

Un classificador Bayesià és un classificador de patrons basat en teories estadístiques d'aprenentatge. A continuació mostrem el seu funcionament.

6.4.2.1 Aprenentatge Bayesià

L'aprenentatge bayesià calcula la probabilitat de cada hipòtesi de les dades, i realitza prediccions sobre aquestes bases. Es realitzaran prediccions fent servir totes les hipòtesis, ponderades amb les seves probabilitats.

Si D és el conjunt de dades i d el valor observat, la probabilitat de cada hipòtesis s'obté aplicant la regla de *Bayes*:

$$P(h_i \mid d) = \alpha P(d \mid h_i) P(h_i)$$

Suposem que volem fer una predicció sobre una quantitat desconeguda X:

$$P(X \mid d) == \sum_i P(X \mid d, h_i) P(h_i \mid d) = \sum_i P(X \mid h_i) P(h_i \mid d)$$

S'ha fet la assumpció que cada hipòtesi determina una distribució de probabilitats sobre X. La equació mostra que les prediccions son el resultat de ponderar les prediccions sobre les hipòtesis individuals.

6.4.2.2 Hipòtesi MAP

L'aprenentatge Bayesià és gairebé òptim, però requereix grans quantitats de càlcul degut a que l'espai d'hipòtesis és normalment molt gran , o inclús pot ser infinit. En la majoria dels casos, el càlcul del sumatori és intractable. Això ens obligarà a recórrer a mètodes aproximats, o bé mètodes simplificats.

Una aproximació molt habitual consisteix en fer les prediccions basant-nos en la hipòtesi més probable, una h_i tal que maximitzi $P(h_i \mid d)$.

Aquesta simplificació se la anomena màxim a posteriori, o bé hipòtesi MAP. Les prediccions que realitzen aquestes aproximacions són aproximadament Bayesianes, fins al punt que $P(X \mid d) \approx P(X \mid h_{MAP})$. Això és degut al següent: A mesura que arriben més dades, la predicció MAP i la Bayesiana s'acosten, perquè les competidores de la hipòtesi MAP es van fent menys probables. Trobar la hipòtesi MAP és habitualment més senzill que l'aprenentatge Bayesià, ja que només requereix resoldre un problema d'optimització. En l'aprenentatge Bayesià es requeria resoldre un problema d'integració, o bé resoldre un gran sumatori.

6.4.3 Classificador Parzen

Aquest classificador està basat en el histograma de les dades [22] . Estima les densitats de cada classe. Aquest és el seu algoritme:

Entrada: Conjunt de mostres d'entrenament T_s i conjunt de mostres de test T.

1. Seleccionar σ_h tal que: $\sum_{k=1}^{K}\sum_{j=1}^{N_k}\ln\left(\hat{p}(z_{k,j} \mid \omega_k)\right)$ sigui màxim.

2. Estimar la densitat: per a cada mostra z del conjunt de test calcular la densitat de cada classe: $\hat{p}(z \mid \omega_k) = \frac{1}{N_k}\sum_{z_j \in T_k}\frac{1}{\sigma_h^N\sqrt{(2\pi)^N}}\exp\left(-\frac{\|z - z_j\|^2}{2\sigma_h^2}\right)$

3. Classificació: Assignar les mostres de T a la classe amb una probabilitat màxima a posteriori: $\hat{\omega} = \omega_k$ amb $k = \arg\max_{i=1,\ldots,K}\left\{\hat{p}(z \mid \omega_i)\hat{P}(\omega_i)\right\}$

6.4.4 Classificador Backpropagation

El model de xarxa neural habitual que utilitza aquest algoritme consisteix en una xarxa neural amb una capa d'entrada amb tants nodes com entrades tinguem, una capa oculta amb un nombre de nodes variable que dependrà de les característiques del problema, i una capa de sortida amb tants nodes com possibles sortides tinguem. En el nostre cas, com que tenim quatre possibles classes (els quatre telenotícies) tindrà quatre nodes a la última capa. L'algoritme d'entrenament s'anomena Back Propagation (propagació cap a endarrere). Té aquest nom degut al seu funcionament. Inicialment, les entrades es propaguen cap a endavant fins a arribar a la sortida. Després, començant per la última capa, es calcula l'error i la contribució a l'error del sistema per a cada node o neurona. Aquesta contribució es calcula amb propagació cap a endarrere fins als nodes de la primera capa (per això el nom de Back Propagation). A partir de la contribució a l'error global de cada node individualment es modificarà el seu valor amb més o menys quantitat segons aquest valor.

6.4.5 Classificador amb expansió Karhunen-Loève

La expansió Karhunen-Loève és la representació d'un procés estocàstic com a combinació de diverses funcions ortogonals. És un concepte semblant al de les sèries de Fourier, però aquí els coeficients són variables aleatòries, i la expansió base depèn del

procés, enlloc de basar-se en senyals sinusoïdals. Les funcions base depenen de la funció de covariància del procés.

Aquesta transformació, anomenada KLT, determinarà una sèrie de coeficients que resultaran útils per a la classificació basant-nos en aquestes dades, alhora que reduiran la dimensionalitat de les dades.

En el cas d'un procés estocàstic centrat {Xt}t [a,b] (on centrat significa que les expectatives E(Xt) estan definides a 0 per tot t) que satisfaci una condició de continuïtat tècnica, admetrà una descomposició de la forma:

$$\mathbf{X}_t = \sum_{k=1}^{\infty} \mathbf{Z}_k e_k(t).$$

On $\mathbf{Z}_k$ són parelles correlatives de variables aleatòries, i les funcions e_k són continues amb valors reals a [a,b] el qual es una parella ortogonal a $L^2[a, b]$.
El cas general del procés que no està centrat, pot ser representat expandint la funció d'expectació (la qual es una funció no aleatòria) en la base e_k .

6.4.6 Classificador amb PCA

PCA és definit com a una transformació lineal ortogonal que transforma les dades en un nou sistema de coordenades on la variància màxima per qualsevol projecció de les dades s'estableix en la primera coordenada, anomenada el primer PCA (*Principal Component Analysis*), la segona màxima variància s'estableix en la segona coordenada, etc.

La transformació PCA equival a la transformació discreta de Karhunen-Loève.

6.4.7 Classificador logístic

Són classificadors basats en el model de regressió estadística per variables dependents amb distribució de *Bernoulli*.

La distribució logística està definida de la següent manera:

$$P(x) = \frac{e^{(x-m)/b}}{|b|[1+e^{(x-m)/b}]^2}$$

6.4.8 Classificador Least Squared Error

Aquest classificador sorgeix com a alternativa al perceptró, ja que aquest darrer no funciona bé en els casos separables. Si el conjunt d'entrenament no és separable, aleshores el procediment iteratiu tendirà a fluctuar al voltant de cert valor [22].

Aquest classificador fa servir una sèrie de vectors K-dimensionals, cadascun associat a una mostra de dades. Calcula els pesos d'aquests vectors mitjançant el criteri de least squares. La solució serà el valor que minimitzi el Least Squared Error.

6.4.9 Classificador amb mixtura de Gaussianes

Aquest classificador [22] , basat en el K veí més proper, assumeix que els objectes en cadascun dels K conjunts existents estan distribuïts amb una distribució gaussiana. Cada conjunt està caracteritzat per la seva mitjana aritmètica (μ_k) i per la seva matriu de covariància (C_k):

$$p(z) = \sum_{k=1}^{K} \pi_k N(z \mid \mu_k, C_k)$$

6.4.10 Classificador K-veí més proper

Aquests classificadors es basen en els models de veïns més propers [17] . Es basen en el següent: és probable que les propietats d'un punt d'entrada particular x siguin similars a les dels punts propers a x.

En aquests mètodes fa falta especificar exactament què és el que s'entén per veí. Si el veïnatge és massa petit, aleshores no contindrà cap punt. En canvi, si és massa gran, pot ser que inclogui tots els punts de dades. La solució a aquest problema consisteix en definir un veïnatge suficientment gran com per a incloure k punts, on k és prou gran com per a assegurar una estimació amb significat. Per a un k fix, la mida del veïnatge varia segons la distribució de les dades. Si les dades estan disperses, el veïnatge serà gran; en canvi, si les dades estan molt properes el veïnatge serà petit.

És evident que ens farà falta una mètrica per a mesurar la distància entre dos punts. La distància euclidiana no és sempre la millor opció. Quan cada dimensió de l'espai es mesura de forma diferent la distància euclidiana no és el més adequat, perquè canviar una escala d'una dimensió podria canviar el conjunt de veïns més propers. Un exemple seria la alçada i el pes.

La solució a aquest problema consisteix a estandarditzar la escala de cada dimensió. Això es fa de la següent manera: mesurem la desviació estàndard de cada característica sobre el conjunt de les dades i expressem els valors com a múltiples de la desviació estàndard d'aquesta característica.

L'aprenentatge supervisat mitjançant aquesta tècnica es fa de la següent manera:

Donat un exemple de test amb entrada *x*, la sortida $y = h(x)$ s'obtindrà a partir dels valors y dels k veïns més propers a x.

Hem de diferenciar el cas simple i el cas continu:

En el cas simple, es pot obtenir la predicció mitjançant el vot de la majoria.

En el cas continu, calculem la mitjana dels k valors, o bé calculem la regressió lineal, ajustant un hiperplà a k punts i predient el valor de x a partir d'aquest hiperplà.

El classificador és simple [22]. La classe assignada a un vector z és la classe amb el màxim nombre de vots de k mostres properes a z.

Si k_k denota el nombre de mostres trobades a la classe w_k, aleshores:

$$\hat{\omega}(z) = \omega_k \qquad \text{amb} \qquad k = \arg\max_{i=1,\ldots,K} \{k_i\}$$

6.4.11 Xarxa neural Levenberg-Marquardt

Aquesta xarxa utilitza un algoritme basat en el mètode de Newton. És un algoritme iteratiu d'optimització en el que el mètode d'iteració està lleugerament modificat respecte l'original. Se'n obté un bon rendiment en l'entrenament de xarxes neurals on el rendiment de la xarxa estigui determinat per l'error mig quadràtic

Interpola entre l'algorisme de Gauss-Newton i el mètode del gradient. LMA és més robust que no pas el GNA, el que significa que en molts casos troba una solució tot i que comenci força lluny del mínim final. Per altra banda, per funcions que tenen un punt de començament raonable, és un pel més lent que el GNA.

La principal aplicació és en el problema de l'encaix de la corba quadràtica. Donat un conjunt de parelles de dades empíriques (t_i, y_i), optimitzar els paràmetres **p** del model de la corba $f(t|\mathbf{p})$ fent així que la suma dels quadrats de les derivades esdevingui mínim.

$$S(\mathbf{p}) = \sum_{i=1}^{m} [y_i - f(t_i \mid \mathbf{p})]^2$$

6.4.12 Xarxa neural Radial Basis

Aquesta xarxa fa servir funcions radials com a funcions d'activació, és a dir, funcions amb valors reals que depenen únicament de la distància a l'origen.

L'estructura típica consisteix en tres capes. Una primera capa d'entrada, una capa oculta amb funcions d'activació radials no lineals i una capa de sortida.

Un vector X d'entrada és usat com a entrada a les funcions radials, cadascuna amb paràmetres diferents. La sortida de la xarxa consisteix en una combinació lineal de les sortides de les funcions radials.

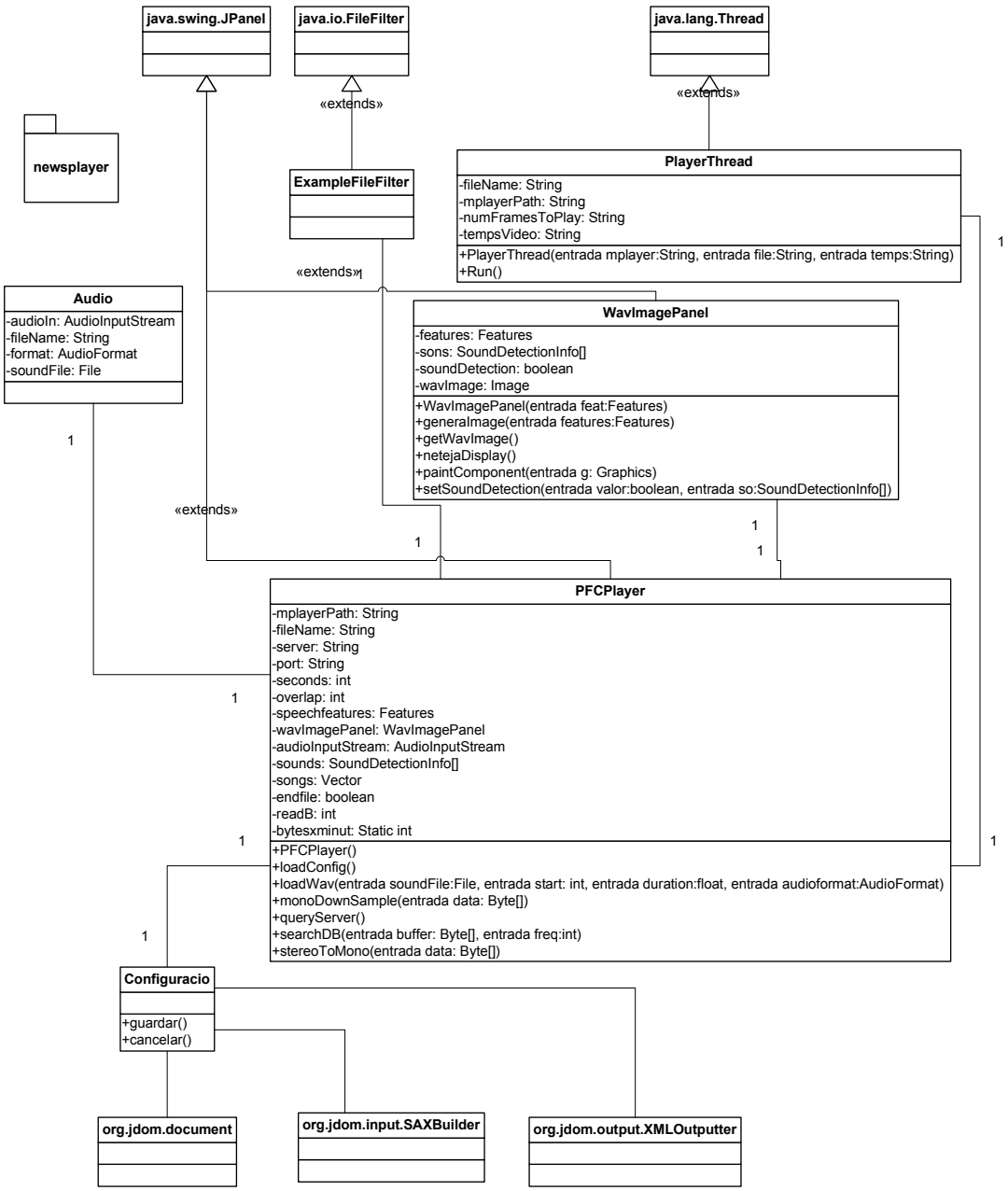
Qualsevol funció φ que satisfaci la propietat $\varphi(\boldsymbol{x})=\varphi(\|\boldsymbol{x}\|)$, es una funció radial. La distancia és normalment la distancia euclidiana.

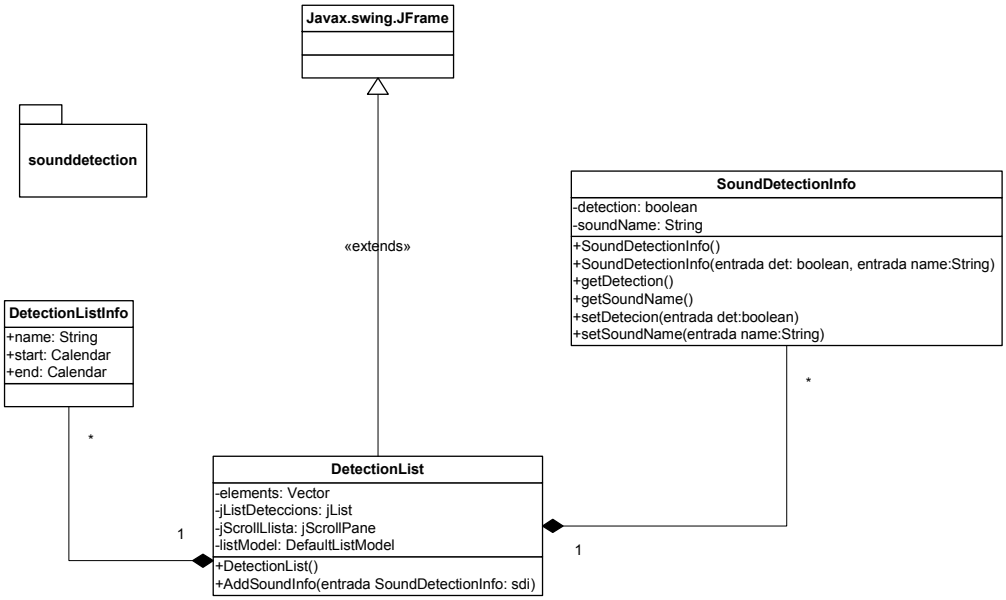
Les funcions Radial Basis són típicament usades per a construir aproximacions de funcions de la forma:

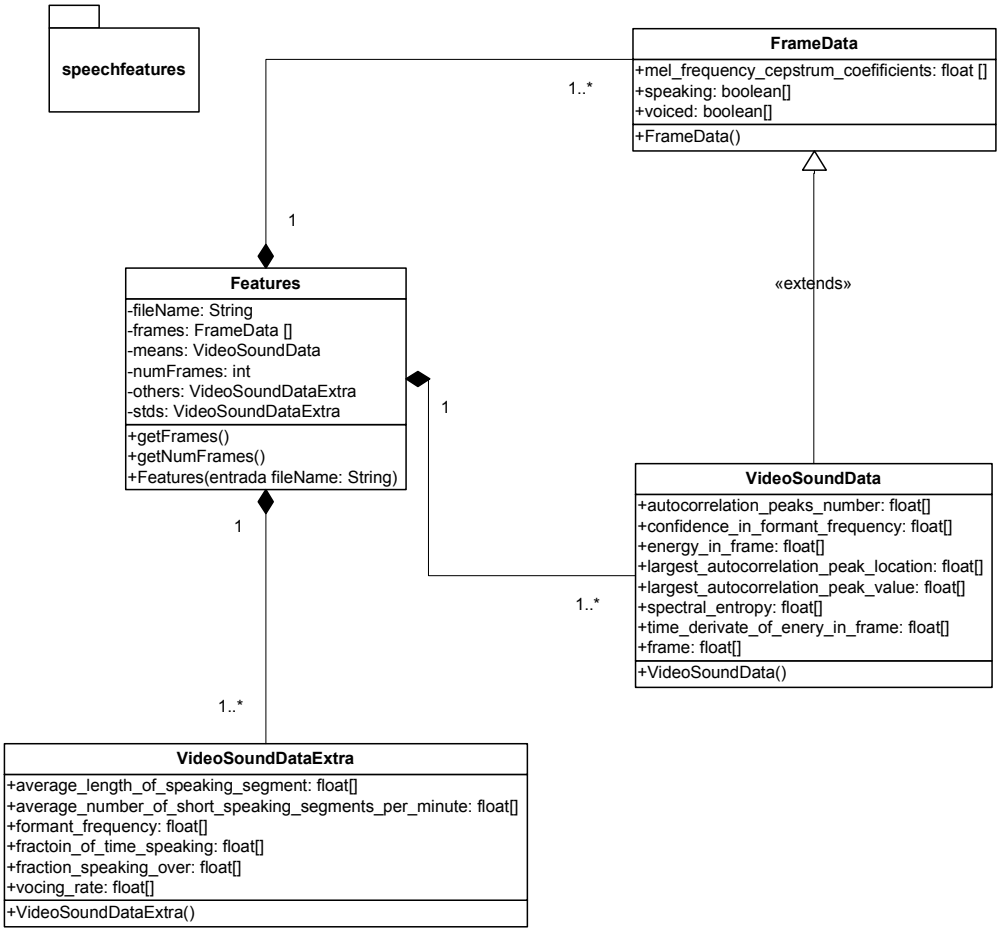
$$y(\mathbf{x}) = \sum_{i=1}^{N} w_i \, \phi(\|\mathbf{x} - \mathbf{c}_i\|),$$

On la funció y(x) aproximada és representada com la suma de N funcions radial basis, cada una associada amb centres diferents $\boldsymbol{c}_i$, i balancejats per un coeficient w_i. Aquest tipus de funció han estat particularment usats en prediccions de series de temps i en el control de sistemes no lineals que mostren un comportament caòtic simple.

6.5 Diagrama UML de classes de la aplicació PFC Player




Javax.swing.JFrame
sounddetection
«extends»
SoundDetectionInfo
-detection: boolean
-soundName: String
+SoundDetectionInfo()
+SoundDetectionInfo(entrada det: boolean, entrada name:String)
+getDetection()
+getSoundName()
+setDetecion(entrada det:boolean)
+setSoundName(entrada name:String)
DetectionListInfo
+name: String
+start: Calendar
+end: Calendar
*
*
1
1
DetectionList
-elements: Vector
-jListDeteccions: jList
-jScrollLlista: jScrollPane
-listModel: DefaultListModel
+DetectionList()
+AddSoundInfo(entrada SoundDetectionInfo: sdi)


speechfeatures
FrameData
+mel_frequency_cepstrum_coefificients: float []
+speaking: boolean[]
+voiced: boolean[]
+FrameData()
1..*
1
«extends»
Features
-fileName: String
-frames: FrameData []
-means: VideoSoundData
-numFrames: int
-others: VideoSoundDataExtra
-stds: VideoSoundDataExtra
+getFrames()
+getNumFrames()
+Features(entrada fileName: String)
1
1
VideoSoundData
+autocorrelation_peaks_number: float[]
+confidence_in_formant_frequency: float[]
+energy_in_frame: float[]
+largest_autocorrelation_peak_location: float[]
+largest_autocorrelation_peak_value: float[]
+spectral_entropy: float[]
+time_derivate_of_enery_in_frame: float[]
+frame: float[]
+VideoSoundData()
1..*
1..*
VideoSoundDataExtra
+average_length_of_speaking_segment: float[]
+average_number_of_short_speaking_segments_per_minute: float[]
+formant_frequency: float[]
+fractoin_of_time_speaking: float[]
+fraction_speaking_over: float[]
+vocing_rate: float[]
+VideoSoundDataExtra()

6.6 Alfabet fonètic internacional

THE INTERNATIONAL PHONETIC ALPHABET (2005)

CONSONANTS (PULMONIC)

	Bilabial	Labio-dental	Dental	Alveolar	Post-alveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Epi-glottal	Glottal
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ			
Plosive	p b			t d		ʈ ɖ	c ɟ	k g	q ɢ		ʡ	ʔ
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	H ʢ	h ɦ
Approximant		ʋ		ɹ		ɻ	j	ɰ				
Trill	ʙ			r					ʀ		ʜ	
Tap, Flap		ⱱ		ɾ		ɽ					ʡ	
Lateral fricative				ɬ ɮ								
Lateral approximant				l		ɭ	ʎ	ʟ				
Lateral flap				ɺ								

Where symbols appear in pairs, the one to the right represents a modally voiced consonant, except for murmured *ɦ*.
Shaded areas denote articulations judged to be impossible. Light grey letters are unofficial extensions of the IPA.

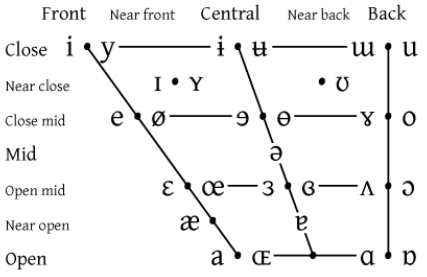
CONSONANTS (NON-PULMONIC)

Anterior click releases (require posterior stops)	Voiced implosives	Ejectives
ʘ Bilabial fricated	ɓ Bilabial	’ *Examples:*
ǀ Laminal alveolar fricated (“dental”)	ɗ Dental or alveolar	pʼ Bilabial
ǃ Apical (post)alveolar abrupt (“retroflex”)	ʄ Palatal	tʼ Dental or alveolar
ǂ Laminal postalveolar abrupt (“palatal”)	ɠ Velar	kʼ Velar
ǁ Lateral alveolar fricated (“lateral”)	ʛ Uvular	sʼ Alveolar fricative

CONSONANTS (CO-ARTICULATED)

ʍ Voiceless labialized velar approximant
w Voiced labialized velar approximant
ɥ Voiced labialized palatal approximant
ɕ Voiceless palatalized postalveolar (alveolo-palatal) fricative
ʑ Voiced palatalized postalveolar (alveolo-palatal) fricative
ɧ Simultaneous *x* and *ʃ* (disputed)
k͡p t͡s Affricates and double articulations may be joined by a tie bar

VOWELS



Vowels at right & left of bullets are rounded & unrounded.

SUPRASEGMENTALS

ˈ Primary stress ˌ Secondary stress [ˌfoʊnəˈtɪʃən]
eː Long eˑ Half-long
e Short ĕ Extra-short
. Syllable break ‿ Linking (no break)

INTONATION

| Minor (foot) break
‖ Major (intonation) break
↗ Global rise ↘ Global fall

TONE

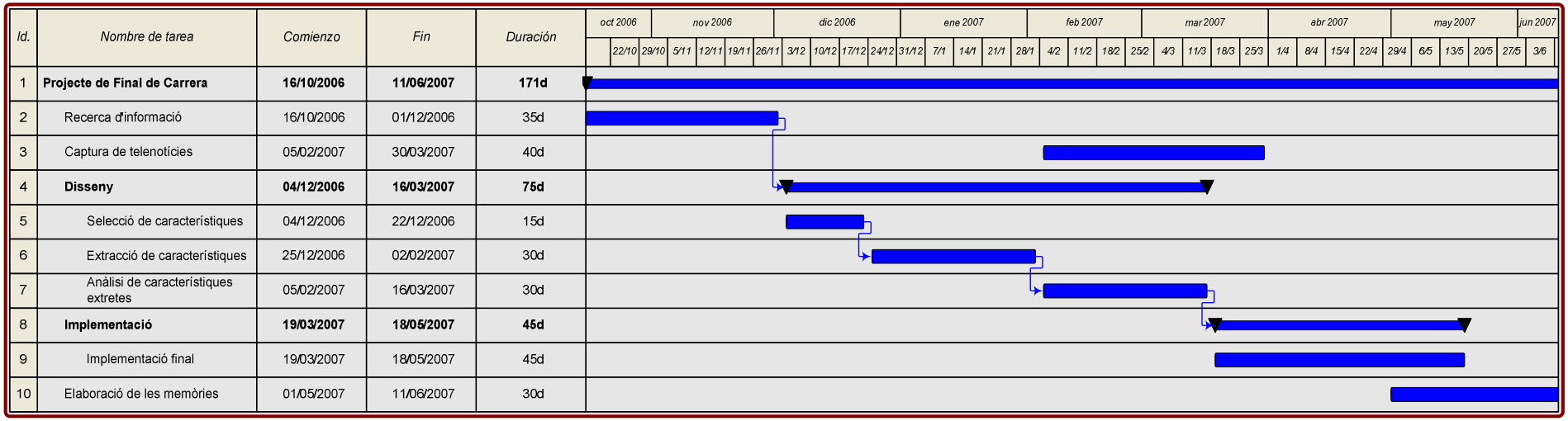
Level tones			*Contour-tone examples:*	
e̋	˥	Top	ě	˨˩ Rising
é	˦	High	ê	˥˨ Falling
ē	˧	Mid	ě	˧˥ High rising
è	˨	Low	ȅ	˩˨ Low rising
ȅ	˩	Bottom	ê	˥˩ High falling
Tone terracing			ē	˧˨ Low falling
↑		Upstep	ě	˨˧˥ Peaking
↓		Downstep	ě	˥˧˨ Dipping

DIACRITICS Diacritics may be placed above a symbol with a descender, as *ŋ̊*. Other IPA symbols may appear as diacritics to represent phonetic detail: *tˢ* (fricative release), *bʱ* (breathy voice), *ʔa* (glottal onset), *ᵊ* (epenthetic schwa), *oᵊ* (diphthongization).

SYLLABICITY & RELEASES		PHONATION		PRIMARY ARTICULATION		SECONDARY ARTICULATION			
n̩ ɹ̩	Syllabic	n̥ d̥	Voiceless *or* Slack voice	t̪ b̪	Dental	tʷ dʷ	Labialized	ɔ̹ χ̹	More rounded
e̯ o̯	Non-syllabic	s̬ d̬	Modal voice *or* Stiff voice	t̺ d̺	Apical	tʲ dʲ	Palatalized	ɔ̜ χ̜ʷ	Less rounded
tʰ ʰt	(Pre)aspirated	n̤ a̤	Breathy voice	t̻ d̻	Laminal	tˠ dˠ	Velarized	ẽ z̃	Nasalized
dⁿ	Nasal release	n̰ a̰	Creaky voice	u̟ t̟	Advanced	tˤ dˤ	Pharyngealized	ɚ ɝ	Rhoticity
dˡ	Lateral release	n̼ a̼	Strident	i̠ t̠	Retracted	ɬ ʒ	Velarized or pharyngealized	e̙ o̙	Advanced tongue root
t̚	No audible release	n̪ d̪	Linguolabial	ä j̈	Centralized	ŭ	Mid-centralized	e̘ o̘	Retracted tongue root
e̞ β̞	Lowered (*β̞* is a bilabial approximant)			e̝ ɹ̝	Raised (*ɹ̝* is a voiced alveolar non-sibilant fricative)				

[26]

6.7 Planificació temporal del projecte



Id.	Nombre de tarea	Comienzo	Fin	Duración
1	**Projecte de Final de Carrera**	**16/10/2006**	**11/06/2007**	**171d**
2	Recerca d'informació	16/10/2006	01/12/2006	35d
3	Captura de telenotícies	05/02/2007	30/03/2007	40d
4	**Disseny**	**04/12/2006**	**16/03/2007**	**75d**
5	Selecció de característiques	04/12/2006	22/12/2006	15d
6	Extracció de característiques	25/12/2006	02/02/2007	30d
7	Anàlisi de característiques extretes	05/02/2007	16/03/2007	30d
8	**Implementació**	**19/03/2007**	**18/05/2007**	**45d**
9	Implementació final	19/03/2007	18/05/2007	45d
10	Elaboració de les memòries	01/05/2007	11/06/2007	30d

Aquesta ha estat la planificació realitzada en la fase inicial del projecte. Hem intentat cenyir-nos a aquesta planificació, i les variacions en dates que s'hagin pogut produir hem procurat que siguin les mínimes per a la correcta realització d'aquest treball.

La fase inicial del projecte va consistir en la recerca de la informació necessària per a dur a terme aquest projecte. Aquesta recerca va consistir recopilar informació sobre les diverses investigacions referents a l'àudio realitzades recentment.

La fase de disseny va consistir en analitzar la múltiple informació trobada. Una vegada analitzada tota aquesta informació, es van seleccionar les eines a utilitzar i les característiques a extreure de l'àudio.

Finalment, la implementació ha consistit en la realització de les diferents aplicacions i experiments per aconseguir els nostres propòsits.

Sergi Espinar Fernández

Bellaterra, 13 de juny del 2007

Aquest és un projecte sobre la indexació de continguts televisius; és a dir, el procés d'etiquetatge de programes televisius per a facilitar cerques segons diferents paràmetres.

El món de la televisió està immers en un procés d'evolució i canvis gràcies a la entrada de la televisió digital. Aquesta nova forma d'entendre la televisió obrirà un gran ventall de possibilitats, permetent la interacció entre usuaris i emissora.

El primer pas de la gestió de continguts consisteix en la indexació dels programes segons el contingut. Aquest és el nostre objectiu. Indexar els continguts televisius de manera automàtica mitjançant la intel·ligència artificial

Este proyecto trata sobre la indexación de contenidos televisivos; es decir, el proceso de etiquetaje de programas televisivos para facilitar búsquedas según diversos parámetros.

El mundo de la televisión se encuentra inmerso en un proceso de evolución y cambios debido a la entrada de la televisión digital. Esta nueva forma de entender la televisión abrirá un gran abanico de posibilidades, permitiendo una interacción entre usuarios i emisora.

El primer paso de esta gestión de contenidos consiste en la indexación de los programas según su contenido. Éste es nuestro objetivo. Indexar los contenidos televisivos de manera automática mediante la inteligencia artificial.

This project is about TV content indexing; that is to say, the TV program labeling process to make searching easier according to different parameters.

TV world is now immersed in an evolution and changing process due to the appearance of the digital TV. This new way of understanding the television will open a new set of possibilities, allowing an interaction between users and digital video broadcasters.

The first step in this process of content management consists in program indexing according to its content. This is our main objective. Indexing TV contents automatically using the Artificial Intelligence.