

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

SOFTWARE PARA CONTABILIZAR AUTOMÁTICAMENTE LA PUBLICIDAD TRANSMITIDA EN RADIO O TELEVISIÓN, MEDIANTE EL RECONOCIMIENTO DE AUDIO

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

ORRALA MORENO JUAN ALFREDO
juanorralla@hotmail.com

DIRECTOR: Ing. VELARDE GUEVARA JAIME EDISON
jaime.velarde@epn.edu.ec

Quito, Septiembre 2012

DECLARACIÓN

Yo, Juan Alfredo Orrala Moreno, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Juan Alfredo Orrala Moreno

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Juan Alfredo Orrala Moreno, bajo mi supervisión.

Ing. Jaime Velarde Guevara
DIRECTOR DEL PROYECTO

AGRADECIMIENTO

A mi papá, Julio, que desde niño siempre aupó mis sueños. A mi mamá, Hilda, por comprender y aceptar el camino optado.

A mis hermanos Mhari, Rhoci, Carhmen, Thoño, Jhosé, Jhulio, Yholi, Jhorge y Mharkos por haberme apoyado en diferentes momentos y circunstancias.

Especialmente a mi hermana Rhoci y mi cuñado Marco por cobijarme en su hogar y poder culminar con total tranquilidad este proyecto.

A los amistades que el destino me brindo y me supieron alentar en la culminación de este trabajo Cristian, Diego y Julio.

Al Ingeniero Jaime Velarde por guiar la elaboración de este proyecto.

Jhuan

DEDICATORIA

A mi papá por inculcarme trabajar con lo posible y soñar con lo imposible; a mi mamá por infundirme a buscar confianza espiritual.

A mis hermanos y mis sobrinos.

Jhuan

RESUMEN

Este trabajo ha sido realizado con el fin de dar una opción al monitoreo manual de publicidad. En ese ambiente el proyecto detecta cuantas veces un determinado anuncio publicitario es reproducido en los medios de radio o televisión. Mediante el desarrollo de un software se apunta a dos objetivos: encontrar la publicidad y emplear el menor tiempo posible en ello.

En el primer capítulo se exponen las causas que originan el desarrollo de este trabajo, los objetivos y un resumen de la manera en cómo se desarrollará la solución al problema planteado revisando teoría y tecnología inmersa al proyecto.

En el capítulo dos se desarrolla el diseño, mostrando la perspectiva de los elementos físicos necesarios para el completo funcionamiento del sistema. Así también, se exponen los requerimientos técnicos del sistema, el procedimiento de cómo resuelve los problemas planteados y las características de los resultado.

En el capítulo tres se realizan pruebas del diseño, explicando gráficamente cada paso que se realiza para obtener un resultado final. Se expone con tablas .los elementos que son utilizados para los reiterados ensayos. Los resultados también son mostrados en este apartado.

Las conclusiones y recomendaciones son tomadas en cuenta en el capítulo cuatro, anticipando que las recomendaciones llevan un lenguaje más técnico, puesto que las personas interesadas en el desarrollo posterior del sistema deben tener conocimiento afín.

PRESENTACIÓN

El monitoreo de publicidad tiene dos propósitos fundamentales, uno interno y otro externo. El primer propósito es tener información lo más precisa posible que ayude a controlar recursos que ayuden a ser más efectiva la tarea de promocionar algún proyecto. El segundo motivo tiene propósitos legales, las empresas comerciales establecidas en Ecuador, deben rendir cuentas al estado de sus ingresos y egresos. Dentro de los egresos, puede presentar inversiones en publicidad impresa, radial o televisada.

Para ambos casos dicha información se puede obtener revisando los registros de tres fuentes: del anunciante, del medio o de un monitoreador. El anunciante, es la firma comercial que contrata el espacio publicitario en una estación de Radio o TV, que vendría a ser el medio. El monitoreador es una empresa especializada en auditar spot publicitarios. La empresa auditora registra, revisa, clasifica y presenta un informe de un determinado comercial transmitido en Radio o TV.

Actualmente los procesos para presentar un informe de una compañía auditora de spot en TV, son relativamente lentos, específicamente en la revisión y clasificación. Para la revisión, se graba de manera digital toda la programación diaria de cada estación de TV (hace mas de 3 años aún se registraba en cintas de VHS), 24 horas después recién se revisa el archivo de video para empezar la clasificación manual de los comerciales.

Entre los datos que se anotan están: canal de TV, firma comercial, nombre del comercial, fecha de la programación, hora de exposición, tiempo de duración, calidad de transmisión, número de aparición, etc. La organización de los detalles de una programación de 20 horas diarias de un canal de TV, la realiza una persona en aproximadamente 4 horas.

En consecuencia, para presentar un informe mensual de una determinada marca publicitaria en un solo canal, se necesita alrededor de 120 horas de trabajo.

Con el avance tecnológico, las empresas monitoreadoras tienen la expectativa de lograr dos de los objetivos, que podría parecer uno solo: contar la publicidad en el

menor tiempo posible. El prototipo aquí desarrollado, busca cumplir con esas dos expectativas.

Cabe aclarar que alrededor del mundo hay muchas compañías dedicadas a este trabajo y alegan tener, con supuestas demostraciones, software capaz de reconocer los anuncios publicitarios de manera rápida y certera. Los propietarios de dicho software, guardan celosamente las herramientas del “cómo lo hacen”.

Por lo antes expuesto se hace nula la posibilidad de probar algún software comercial, revisar su alcance o realizar comparaciones con el prototipo aquí desarrollado.

1. CAPÍTULO UNO. INTRODUCCION Y FUNDAMENTOS TEORICOS

1.1.INTRODUCCION

En esta sección se proporcionará una introducción al prototipo desarrollado, se incluye la descripción del método para el reconocimiento del audio exponiendo el modelo estadístico matemático. Consta de varias sub-secciones: propósito, ámbito del sistema, definiciones, referencias y visión general del trabajo. Se describe las herramientas computacionales utilizadas y se expone el propósito del software en el área de la publicidad y mercadeo.

1.2.PROPÓSITO

Este trabajo ha sido realizado con el fin de dar una opción al monitoreo manual de publicidad. En ese ambiente es necesario conocer cuantas veces un determinado anuncio publicitario es reproducido en los medios de radio o televisión. Mediante el desarrollo de un software se apunta a dos objetivos: encontrar la publicidad y emplear el menor tiempo posible en ello.

1.3.ÁMBITO DEL SISTEMA

1.3.1. PROYECTO SPOTID

Este proyecto es denominado SpotID, el cual procede de dos palabras Spot (Publicidad en radio o televisión) e ID (Identificación) sugiere que hace el programa.

1.3.2. LO QUE HACE SPOTID

El sistema a desarrollar debe permitir:

- Contabilizar los spots publicitarios de un anunciante a la vez.
- Cargar archivos de audio pregrabados en formato WAV.
- Extracción de audio de archivos de videos pregrabados.
- Extraer el audio de spots.

- Grabar y almacenar los archivos de audio extraídos.
- Registrar las señales transmitidas por TV y radio.
- Presentar informe imprimible de los resultados.

El sistema no estará diseñado para realizar las siguientes tareas:

- Editar los archivos de audio y videos almacenados.
- Reconocer las publicidades en vivo.
- Contar todas las publicidades emitidas en un solo escaneo.

1.3.3. OBJETIVOS

Con el avance tecnológico, las empresas encargadas de monitorear publicidad tienen la expectativa de lograr dos objetivos, el primero es tener información lo más precisa posible que ayude a controlar recursos que ayuden a ser más efectiva la tarea de promocionar algún proyecto. El segundo objetivo tiene propósitos contables, que ayuda rendir cuentas de las inversiones en publicidad impresa, radial o televisada.

1.3.1. BENEFICIOS

Los beneficios destacables de SpotID son:

- Contar la publicidad en el menor tiempo posible.
- Presentar informe detallado con respaldo impreso y audiovisual.
- Conocer los movimientos promocionales de alguna campaña publicitaria.

1.4. FUNDAMENTOS TEORICOS

Cualquier sistema en el que existe interacción de variables de entrada y salida, (Figura 1-1), puede requerir retroalimentación para poder tomar acciones correctivas en busca de una mejor funcionalidad del sistema.

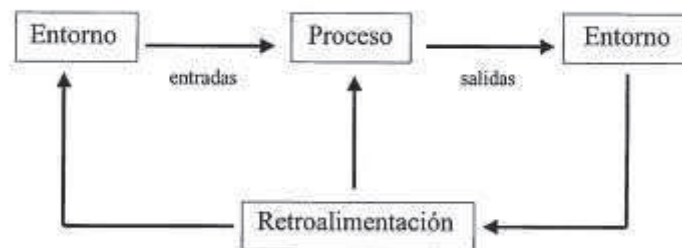


Figura 1-1 Diagrama de bloques de un Sistema en el que interactúan variables Entrada y Salida.

En el área de Publicidad y Mercadeo depende de la retroalimentación para poder optimizar los recursos y maneras de llegar a los “target” (En el ámbito de la publicidad, se utilizan como sinónimo para designar al destinatario ideal de una determinada campaña, producto o servicio.)¹; la retroalimentación puede tener varias fuentes como clientes, canales de comunicación y empresas que monitorean publicidad. SpotID entra en el campo del monitoreo de publicidad pues detecta la aparición de anuncios publicitarios de radio y televisión.

SpotID trabaja con dos señales de audio. La primera señal es una muestra de audio de un anuncio publicitario, Audio Spot (AS); y, la segunda señal de audio es el Audio Programación (AP), que registra el audio de la programación difundida en radio o televisión. Así, SpotID está en capacidad de comprobar si existe una o varias coincidencias del “spot” (pautas publicitarias) a lo largo de toda la programación (Figura 1-2.) lo que permitirá verificar al dueño de la publicidad si los spots contratados han sido publicados en tiempo y espacio convenidos.

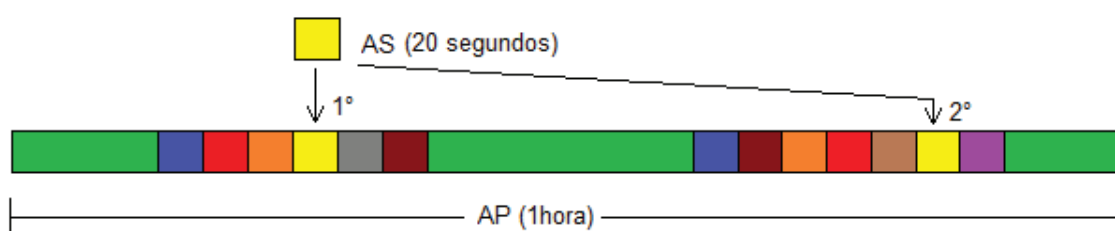


Figura 1-2 Representación dos apariciones de AS en todo el AP

En la actualidad, se realiza manualmente la tarea de buscar varios AS en un AP, por lo que uno de los objetivos de este trabajo es disminuir el tiempo de análisis comparado al trabajo realizado por el ser humano, por tal razón es necesario un

computador con requerimientos mínimos de velocidad de procesamiento. El análisis y presentación de resultados, se muestran en una interfaz comprensible e imprimible, que es desarrollado en lenguaje computacional Matlab®.

La detección de una señal repetida se basa en técnica estadística llamada Correlación Cruzada, que es “una operación matemática que permite cuantificar el grado de similitud entre dos señales, aunque aparentemente no haya evidencias de coincidencia temporal entre ellas”².

Previo al proceso de detección se realiza la adquisición de señales de radio o televisión, mediante hardware capturador de señales de video y audio por ejemplo el Zogis Real Angel 400U PRO, que debe ser instalado en el computador.

El video o el audio serán almacenados en el computador mediante archivos con una nomenclatura fijada por el capturador, lo que permite al SpotID clasificar y exponer de manera detallada el nombre de estación de TV o radio, el canal o frecuencia, la fecha de grabación, la hora del día en que inicia y termina la grabación y el tipo de horario (clasificación AAA, AA ó A).

1.5.DEFINICION DE PUBLICIDAD

Según Iván Thompson, expone: “La publicidad es una forma de comunicación impersonal y de largo alcance que es pagada por un patrocinador identificado (empresa lucrativa, organización no gubernamental, institución del estado o persona individual) para informar, persuadir o recordar a un grupo objetivo acerca de los productos, servicios, ideas u otros que promueve, con la finalidad de atraer a posibles compradores, espectadores, usuarios, seguidores u otros.”³

Por lo tanto, si la publicidad involucra una inversión de dinero, implica que debe ser controlada.

1.5.1. CONTROL PUBLICITARIO

Tomando como ejemplo publicitario, se puede decir que, si una Empresa X planea hacer conocer su producto estrella por televisión a un target determinado con una

secuencia de apariciones estratégicamente establecidas lo largo del día; y, como la Empresa X no es dueña de la Estación Televisiva en donde se publica su producto estrella, no sabrá directamente cuán conveniente será la decisión de promocionar el producto con esas condiciones descritas.

El control publicitario busca argumentos para determinar cuán rentable es la campaña de promoción (pautaje). Con esta información, el creador del mensaje publicado tomará correctivos cualitativa y cuantitativamente para que mantenga o aumente la rentabilidad de la inversión publicitaria. Este control genera información luego de un monitoreo de la publicidad.

1.5.2. MONITOREO PUBLICITARIO⁴

El monitoreo de medios publicitarios es un recurso indispensable para cualquier entidad que quiera dar seguimiento de determinada estrategia publicitaria con el fin de obtener:

Verificación de Cumplimiento de Contrato Publicitario.

Información de Actividad de Competidores.

Respaldo de información Propia y de Competidores.

1.5.2.1. Verificación de Cumplimiento de Contrato Publicitario:

La verificación de la transmisión de todos los spots contratados se puede realizar a través de un software de reconocimiento automático de spots en los canales de televisión y radio de una cobertura determinada; como resultado se genera un informe que puede detallar si se cumplen con algunos conceptos básicos de calidad.

Con el informe, el dueño de los spots, estará al tanto de posibles anomalías en la transmisión, tales como:

Spots no transmitidos.

Spots transmitidos fuera del horario contratado.

Spots transmitidos de diferente versión a la contratada.

1.5.2.2. Información de Actividad de Competidores:

El reconocimiento automático de Spots permite obtener información de anunciantes en medios televisivos y radiales, de tal manera que se pueda acceder a reportes de desglosados, donde se podrá detallar datos respecto a anunciantes de la competencia como:

Actividad vigente.

Intensidad de la campaña

Inversión

Tendencias

Secuencia de pauta (días, horarios, programas).

Alcance de publicidad (local, regional o nacional).

1.5.2.3. Respaldo de Información Propia y de Competidores.

Al tener el contenido audiovisual de manera digital, permite almacenar y distribuir la copia de los diferentes spots publicitarios emitidos diaria, semanal o mensualmente.

1.6. SEÑALES DIGITALES

A cualquier señal analógica (luz, sonido, temperatura) se puede realizar un Procesamiento Digital de Señales (PDS) en un computador, pero antes a la señal análoga se la debe codificar de forma digital; en el caso de este trabajo las señales analógicas a tratar son audio y video.



Figura 1-3 Sistema de Conversión Analógica a Digital

El acondicionamiento de la señal analógica la transforma en una secuencia de códigos numéricos en tiempo y amplitud. Este procedimiento se lo conoce como

Conversión Analógica-Digital (CAD), Figura 1-3, y se cumple este propósito realizando el muestreo y la cuantización.

Una vez digitalizadas podrán ser reproducidas realizando el proceso inverso de Conversión Digital-Analógico (CDA).

1.6.1. MUESTREO

Muestreo es la adquisición instantánea de la señal analógica en el tiempo y es el momento en que se toma una muestra de la señal, en este caso señal de audio. Este proceso de particionar en el tiempo la señal analógica es llamado “discretización en el tiempo”.

En la Figura 1-4, muestra que en un determinado tiempo T , y solo en ese momento, se toma una imagen de la señal analógica.

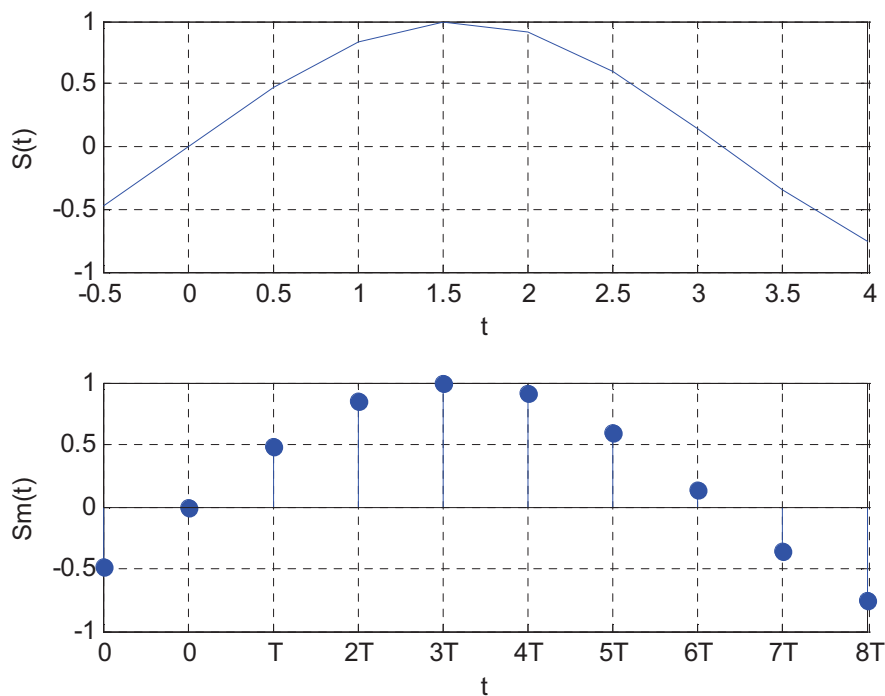


Figura 1-4 Muestreo de una Señal Analógica

El muestreo le da calidad a la señal de audio digital, así, mientras más muestras se tome de la señal original mejor será la calidad del audio. La periodicidad con que se tome una muestra se llama frecuencia de muestreo.

Se puede ver en la Figura 1-5 ejemplos de diferentes tiempos de muestreo, y se concluye que a 12 muestras por ciclo, o sea, a la mayor frecuencia de muestreo, la imagen digital es más fiel a la señal original.

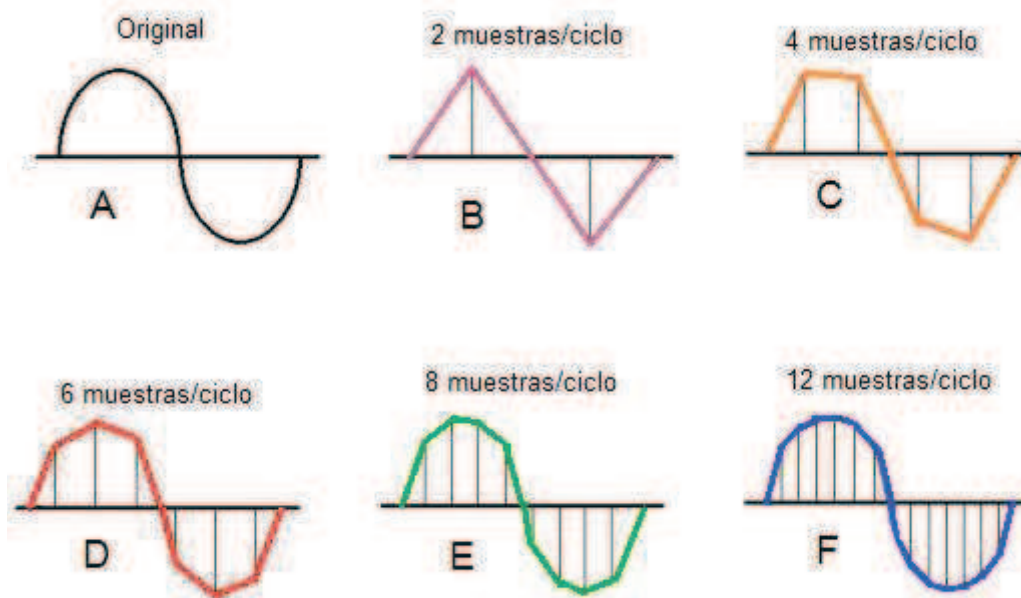


Figura 1-5 Tiempos de Muestreo o Discretización

Según el teorema de Claude Shannon (ingeniero electrónico y matemático estadounidense, recordado como el padre de la “teoría de la información”)⁵, para obtener una buena reconstrucción de una señal digitalizada la frecuencia de muestreo debe ser por lo menos el doble del ancho de banda de la frecuencia de la señal a analógica.

En la Tabla 1.1 se muestra la frecuencia de muestreo con la que se transmite el audio en diferentes medios audibles.

Dispositivo de Audio	Frecuencia de muestreo
Teléfono	8 Khz
Radio FM	22 Khz
CD	44 Khz
DVD	48 Khz

Tabla 1.1 Calidad de frecuencia de muestreo

Por ejemplo, el audio de la voz humana que se escucha a través de la línea telefónica tiene una frecuencia de muestreo de 8Khz, este sonido no se percibe como la voz original, pero se entiende lo que se dice y quien lo dice. Esto sucede debido a que el rango de frecuencia perceptible del el odio humano tiene un ancho de banda alrededor de 20khz, pero para ser mínimamente perceptible y entendible sólo es necesario escuchar un ancho de banda de 4khz; siguiendo el teorema de Shannon se necesitará una frecuencia de muestreo mínima de 8Khz.

Una vez que se decide con qué frecuencia se toman muestras de la señal analógica continúa el proceso de digitalización cuantizando cada muestra con un código binario.

1.6.2. CUANTIZACION

La cuantización establece dos cosas, primero un número determinado de niveles que tendrá la señal digital; y segundo, cómo se codificará cada uno de los niveles.

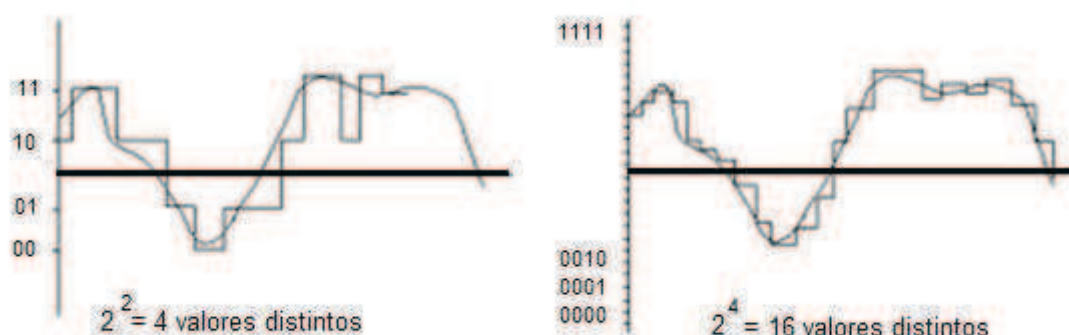


Figura 1-6 Cuantización con 2 y 4 bits

Como muestra la Figura 1-6 la gráfica de la izquierda representa una cuantización realizada con dos bits por muestra, por lo que máximo podrá alcanzar a representar a la señal analógica con 4 valores diferentes, lo que produce un bosquejo digital poco similar a la señal original.

En el gráfico derecho de la Figura 1-6, se aumenta el número de bits a 4 por muestra para cuantizar la señal digital, esto produce un mejor trazo digital. Lo que

lleva a deducir que a mayor número de bits por muestra mayor fidelidad de la señal digital frente a la señal analógica u original.

1.6.3. VELOCIDAD DE BITS

Velocidad de bits se conoce a la cantidad de bits que se han muestreado en un segundo (bps). Si un archivo de audio tiene 8 bits por muestra con una frecuencia de muestreo de 8Khz (8000 muestras por segundo), entonces se tendrá

$$V_{bits} = \frac{8 \text{ bits}}{\text{muestra}} * \frac{8000 \text{ muestras}}{\text{segundo}} = 64000 \frac{\text{bits}}{\text{segundo}} = 64\text{kbps} \quad (\text{Ec. 1.1})$$

La velocidad de bits determinará el tamaño de los archivos de audio por cada segundo de sonido almacenado.

1.7. FORMATOS DE AUDIO.

El formato es el protocolo de digitalización que establecerá el número de muestras por segundo, los bits y la codificación de los niveles por muestra; por ejemplo MP3, WAV, OGG, etc. son formatos de audio digitalizados.

Independiente de su propósito, existen formatos de archivo de audio digital que al ser reproducido deben ser mínimamente perceptible y entendible. Los formatos de audio digital tienen varias configuraciones para optimizar recursos de memoria el número de bits utilizados hacen que el audio mantenga fidelidad, entiéndase memoria como bits.

1.7.1. FORMATO WAV⁶

WAV es el apócope de WAVEform audio file format, fue desarrollado por IBM y Microsoft. Puede configurar el registro en audios monofónico o estereofónico, la frecuencia de muestreo y la cuantización. La Tabla 1.2 muestra las posibles configuraciones que permite el formato WAV.

Por lo general para cuantizar muestras de audio se usan 8 ó 16 bits por muestra. Un archivo WAV según su configuración de almacenamiento puede ocupar un tamaño desde centenas kilobytes hasta un algo más de 4 Gigabytes.

Modo Canal	Bits/Muestra	Frecuencia de Muestreo (Hz)
		6000
		8000
	2	11025
	3	12000
	4	16000
Mono	5	22050
Estéreo	8	24000
	16	32000
	24	44100
	32	48000
		88200
		96000

Tabla 1.2 Configuraciones permitidas dentro del formato de audio WAV

Por ejemplo un archivo de audio WAV con la velocidad de bits de 64Kbps (64000 bits por segundo) se entenderá que almacenará un minuto de audio en 3840kbps (3.84 Mb). Exponiendo de otro modo, un segundo archivo de audio a 192Kbps (192000 bits por segundo), emplearía 11520Kb (11.52Mb) de memoria.

Archivo	Bits/segundo	Bits/muestra	Muestras/segundo	Tamaño 1 minuto
64 kbps	64000	8	8000	3840kb
192 kbps	192000	8	8000	11520kb

Tabla 1.3 Velocidad de bits

Se deberá entender que el segundo archivo WAV de 192kbps, tendrá mejor calidad acústica que un audio grabado a 64kbps. A su vez, se puede elegir un número de muestras suficientes como para ser perceptible y entendible por el ser humano, que podría ser 8Kbps.

1.8. CAPTURADORES DE VIDEO⁷

Una capturador (o tarjeta sintonizadora) de televisión es un periférico que permite ver los distintos tipos de televisión en la pantalla de ordenador. La visualización se puede efectuar a pantalla completa o en modo ventana. La señal de televisión puede proceder de una antena (externa o portátil) o bien de la emisión de televisión por cable.

Este periférico puede ser una tarjeta de expansión, generalmente de tipo PCI, o bien un dispositivo externo que se conecta al puerto USB. Los modelos externos codifican la grabación por software; es decir, que es el procesador del ordenador es quien realmente hace todo el trabajo. Esto permite que un computador efectúa grabación de la señal. Estas tarjetas también pueden ser usadas para captar señales de alguna fuente de video como cámaras filmadoras, reproductores de DVD o VHS, etc.

Las tarjetas sintonizadoras se distribuyen junto a los drivers y el software que permite la sintonización, almacenamiento, visualización y grabación directa o programada de los canales. También existe software gratuito de terceros que funciona con cualquier tarjeta sintonizadora y que en muchos casos mejora la calidad de la visualización y de la grabación obtenida con el software original de la tarjeta sintonizadora:

Entre los fabricantes de tarjetas sintonizadoras se tiene:

- Dscaler
- Kastor!TV
- MythTV
- Tvtime
- Xawtv
- Zapping
- Zogis

Se ha optado por el capturador de Marca Zogis, debido a lo económico con respecto a productos similares y por la facilidad de montaje y desmontaje del computador debido a su conexión USB.

1.8.1. ZOGIS REAL ANGEL 400U DONGLE USB 2.0 PRO⁸

ZOGIS Real Angel 400U Dongle USB 2.0 PRO es un sintonizador y capturador de señales de televisión analógica, radio FM y video compuesto para trabajar en equipos PC o portátil con interface USB. Incluye Drivers y Software.

El hardware requiere alimentación de 5V, que se proveen por el puerto USB, el mismo que es utilizado para el envío de datos hacia la PC. La entrada de señal básicamente consta de cuatro tipos:

- Por antena de Aire
- Señal de Televisión por Cable,
- Antena para Radio FM; y ,
- Entrada de Video RCA

Se pueden realizar funciones de TV analógica mediante un control remoto por infrarrojos o con la aplicación Microsoft Media Center.



Figura 1-7 ZOGIS Real Angel 400U Dongle USB 2.0 PRO

Cualquier archivo de vídeo se puede convertir desde y hacia avi, mpg, etc con el convertidor de MPEG a través de más de Audio y Video por terminales RCA y Súper Video. También se puede editar archivos con el MPEG Editor. Por último,

se puede hacer una copia del archivo editado en DVD / VCD / SVCD con DVD Burn.

Los drivers están basados en arquitectura Microsoft, es totalmente compatible con las aplicaciones, como Cyberlink PowerCinema, ArcSoft TotalMedia, Windows Media Center.

Mediante VivaTV, que es el software de edición y captura de audio y video del RA 400U, se logra visualizar, escuchar y configurar los modos de grabación y reproducción.

En la configuración de captura se puede establecer formatos de video como AVI, MPEG, H264, DVD, VCD, M101. Además de ajustar la resolución de captura en tamaños de 720x576, 720x480, 640x480, 352x288, 352x240 y 320x240. Con opción a grabar sin audio o solo audio el formato de audio viene pre-establecido con el formato de video que se opte. El formato NTSC o PAL es pre-establecido al momento de elegir el país del usuario.

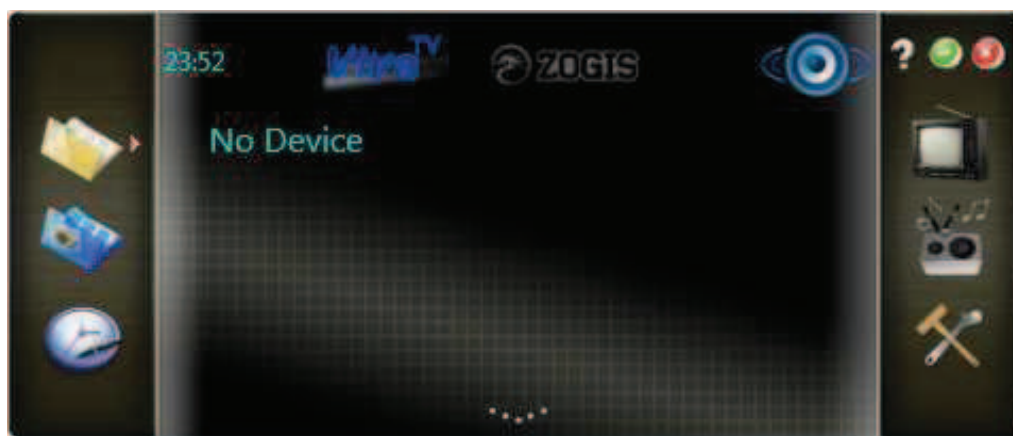


Figura 1-8 Pantalla principal de VivaTV

También existe la opción de programar horarios de grabación semanal, diaria o por horas. Se puede configurar la ruta donde se almacenará el archivo y el nombre.

Para la instalación en el computador es necesario tomar en consideración los siguientes requisitos:

- Intel Pentium-IV 2.0GHz, procesador AMD equivalente o superior
- 512 MB de memoria RAM
- Puerto USB 2.0
- Tarjeta gráfica con 64 MB de memoria (compatible con DirectX 9.0c)
- Tarjeta de sonido compatible con AC97
- 2 GB de espacio libre en el disco duro
- Unidad de CD-ROM (para la instalación del software)
- Microsoft Windows ® XP SP2/XP MCE 2005/Windows Vista

1.9. DVDVIDEOSOFT FREE STUDIO⁹

DVDVideoSoft Free Studio es un suite de programas gratuitos para la edición de archivos de audio y video. Permitiendo realizar un sinnúmero de acciones con estos. De aquí se desprende la subrutina Free Video to MP3 Converter de distribución gratuita.

1.9.1. FREE VIDEO TO MP3 CONVERTER⁵

Es un programa que permite extraer cualquier sonido de un video y guardarlo en formato MP3, M4A, M4R, WAV, WMA, OGG, ACC, FLAC, APE.

Por ejemplo, con Free Video to MP3 Converter se puede extraer efectos de sonido, bandas sonoras, canciones, efectos ambientales e incluso los diálogos de cualquier vídeo, película o serie que se tenga guardado en la PC.

Para extraer el sonido, se debe cargar el video, seleccionar el momento exacto e iniciar el proceso. Luego, Free Video to MP3 Converter generará un archivo WAV con el audio del vídeo. Se puede configurar la frecuencia de muestreo, canales (mono o estéreo)

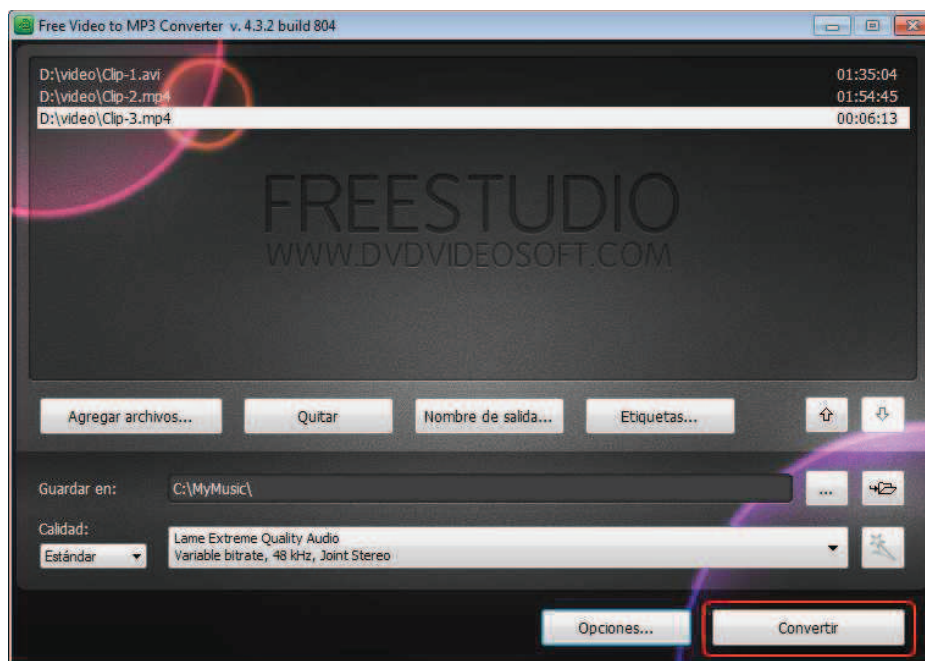


Figura 1-9 Pantalla Principal de Free Video to MP3 Converter

Se puede guardar los vídeos en calidad alta, estándar o económico. Los formatos de entrada: *.avi; *.ivf; *.div; *.divx; *.mpg; *.mpeg; *.mpe; *.mp4; *.m4v; *.webm; *.wmv; *.asf; *.mov; *.qt; *.mts; *.m2t; *.m2ts; *.mod; *.tod; *.vro; *.dat; *.3gp2; *.3gpp; *.3gp; *.3g2; *.dvr-ms; *.flv; *.f4v; *.amv; *.rm; *.rmm; *.rv; *.rmvb; *.ogv; *.mkv; *.ts.

Free Video to MP3 Converter es totalmente gratuito.

1.10. MATLAB®¹⁰

MATLAB es un lenguaje de alto nivel y muestra un entorno interactivo que permite realizar tareas de cálculo complejas de forma más rápida que con los lenguajes de programación tradicionales, como C, C++ y Fortran. Es un software de programación para uso general y puntal en Ingeniería y Ciencias. Trabaja con matrices y es capaz de resolver complejas operaciones matemáticas con pocas líneas de programación, las cuales pueden ser instrucciones, funciones predefinidas por Matlab® o programas desarrollados por el usuario.

Tiene un entorno de Simulación de Sistemas y capaz de interactuar con datos generados por software o hardware externos a Matlab® lo que permite trabajar en

diversos campos como automovilismo, aviación, astronomía, electrónica, medicina, biología, audiovisuales, climáticos, etc.

1.10.1. CORRELACION CRUZADA ¹¹

Matlab utiliza la función $xcorr$ ¹², que calcula la secuencia de correlación cruzada de un proceso aleatorio.

$c = xcorr(x, y)$ devuelve la secuencia de correlación cruzada en un vector de longitud $2*N-1$, donde x e y son vectores de longitud N ($N > 1$). Si x e y no tienen la misma longitud, el vector más corto es rellenado con ceros a la longitud del vector más largo. Por default, $xcorr$ calcula las correlaciones sin normalización.

En procesamiento de señales, la correlación cruzada es una medida de la similitud entre dos señales, frecuentemente usada para encontrar características relevantes en una señal desconocida por medio de la comparación con otra que sí se conoce. Es función del tiempo relativo entre las señales, y tiene aplicaciones en el reconocimiento de patrones y en criptoanálisis.

Se define como correlación cruzada de dos secuencias reales de valores medibles $x[n]$ e $y[n]$, como:

$$r_{xy}[m] = \sum_{n=-\infty}^{\infty} x[n]y[n - m] \quad (\text{Ec. 1.2})$$

Ó, desplazando las muestras en el tiempo, como:

$$r_{xy}[m] = \sum_{n=-\infty}^{\infty} x[n + m]y[n] \quad (\text{Ec. 1.3})$$

Las ecuaciones 1.2 y 1.3 son equivalentes. Nótese que la variable independiente de r_{xy} no es el eje de tiempos n , sino que es una variable m indicadora del desplazamiento relativo entre secuencias $x[n]$ y $y[n]$.

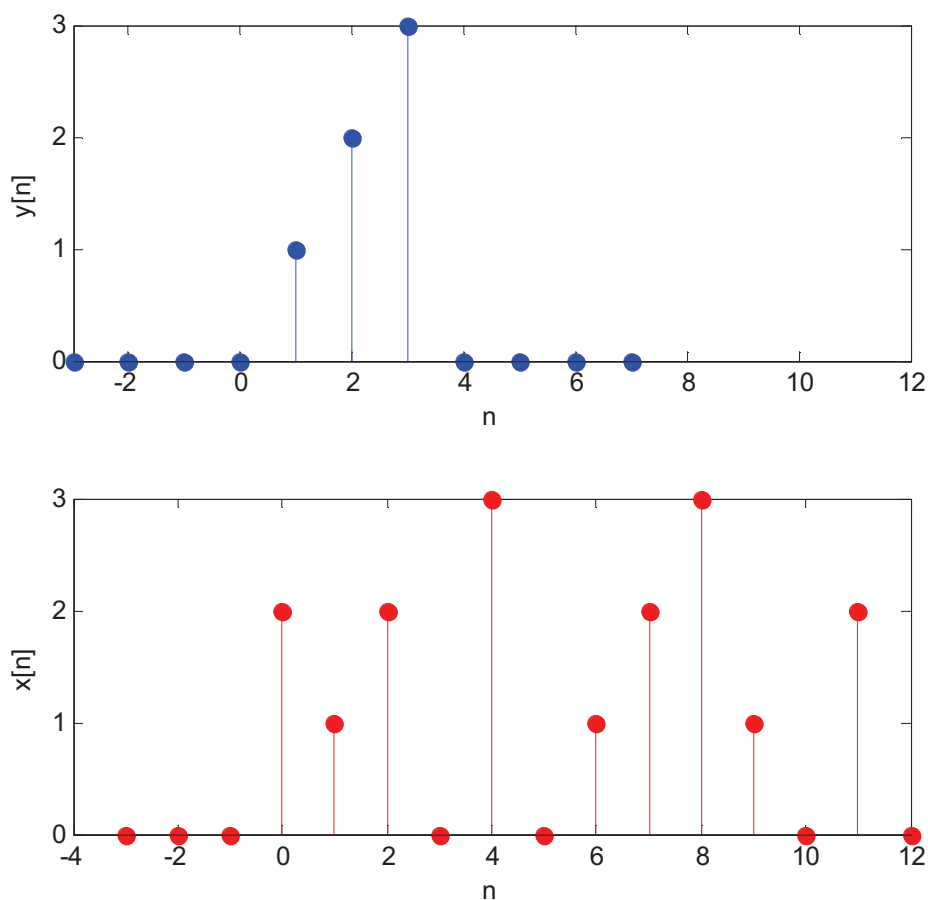


Figura 1-10 Secuencias de Datos X e Y

Comparando la definición de la correlación como una convolución, es inmediato comprobar que:

$$r_{xy}[m] = x[m]y[-m] \quad (\text{Ec. 1.4})$$

Es decir, puede interpretarse la correlación como una convolución en la que se ha invertido el eje de tiempos de uno de los operandos (operación de reflexión).

Ejemplo de cálculo gráfico: Hallar la correlación cruzada $r_{xy}[m]$ entre las dos secuencias que se ilustran en la Figura 1-10.

De forma similar al cálculo gráfico de convoluciones, se va variando m (lo que equivale a ir desplazando la secuencia $x[n+m]$, o la secuencia $y[n+m]$, sobre el eje de tiempos). Figura 1-11.

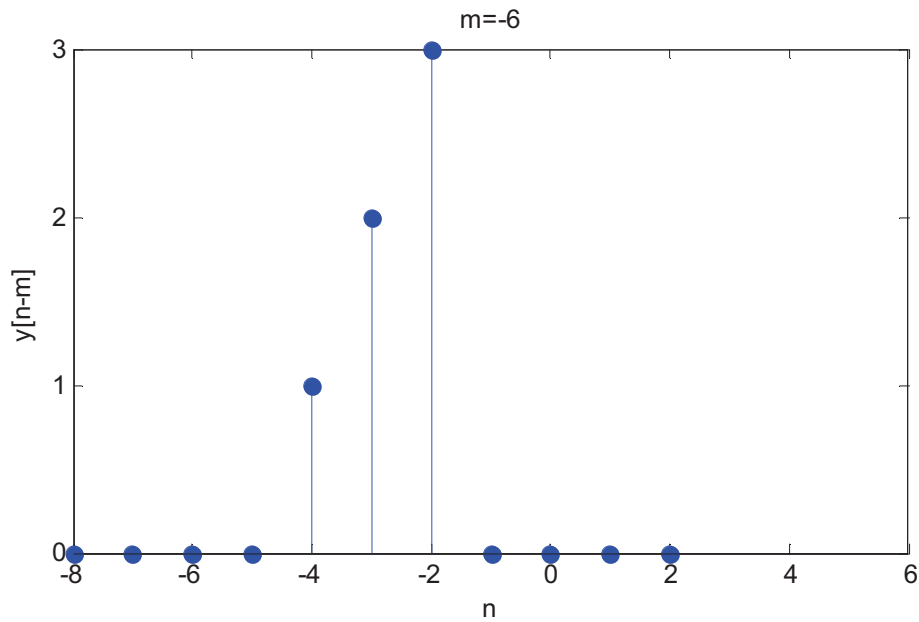


Figura 1-11 Desplazamiento de Y cinco “pasos” de n hacia la izquierda

A diferencia de la convolución, en este caso no hay que invertir el eje de tiempos.

Para cada valor de m la ecuación 1.2

$$r_{xy}[m] = \sum_{n=-\infty}^{\infty} x[n]y[n-m]$$

Nos indica que se suman los productos de las muestras coincidentes en un mismo instante de tiempo $[m]$.

$$r_{xy}[-4] = 0$$

$$r_{xy}[-3] = (2 * 3) = 6$$

$$r_{xy}[-2] = (3 * 1) + (2 * 2) = 7$$

$$r_{xy}[-1] = (2 * 3) = 6$$

Y así sucesivamente, se obtiene

$$r_{xy}[-1] = 10 \quad r_{xy}[6] = 11$$

$$\begin{array}{ll}
 r_{xy}[0] = 5 & r_{xy}[7] = 5 \\
 r_{xy}[1] = 11 & r_{xy}[8] = 7 \\
 r_{xy}[2] = 6 & r_{xy}[9] = 4 \\
 r_{xy}[3] = 6 & r_{xy}[10] = 2 \\
 r_{xy}[4] = 8 & r_{xy}[11] = 0 \\
 r_{xy}[5] = 14 & r_{xy}[12] = 0
 \end{array}$$

El resultado que corresponde a la Figura 1-12 muestra que en el desplazamiento $m=5$, la correlación tiene valor máximo de todas las sumatorias. Lo que indica que habría que desplazar 5 unidades de tiempo a la señal $y[n]$ hacia la derecha para que tener una coincidencia de picos o amplitudes.

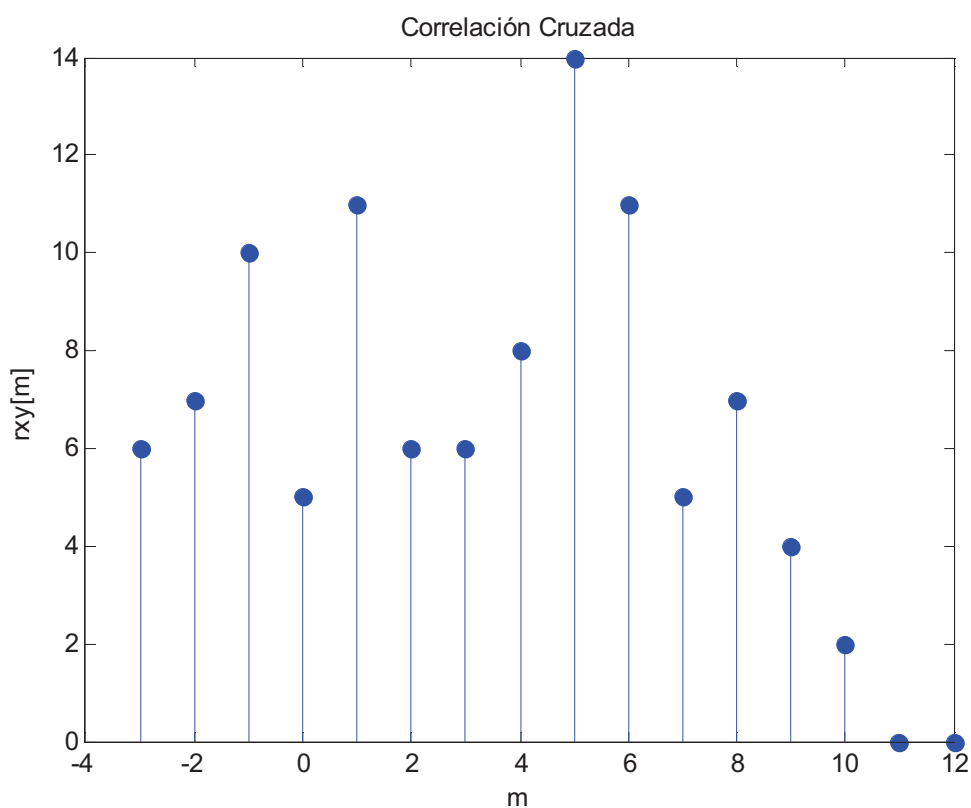


Figura 1-12 Correlación Cruzada de Secuencias X e Y

Un segundo ejemplo, en la Figura 1-13, muestra dos señales senoidales iguales en amplitud pero desfasadas en el tiempo; el valor máxima de la correlación P_0 y P_1 será periódico debido a que en más de una vez se encontrará la misma señal en fase. Lo que significa tener más de una vez repetida

El cálculo de la correlación cruzada entre las dos señales $P0$ y $P1$, permitirá decidir si $P1$ se “asemeja” a la señal $P0$ y en qué posición temporal se encuentra. La señal $P1$ es desplazada en el tiempo retrasando o adelantándola. En el caso de la Figura 1-13, el punto máximo de correlación será cuando coincidan ambas señales en amplitud y fase y será el momento en que $P1$ será muy semejante a la señal $P0$. Además la posición del pico nos dará el desplazamiento de la señal en el tiempo.

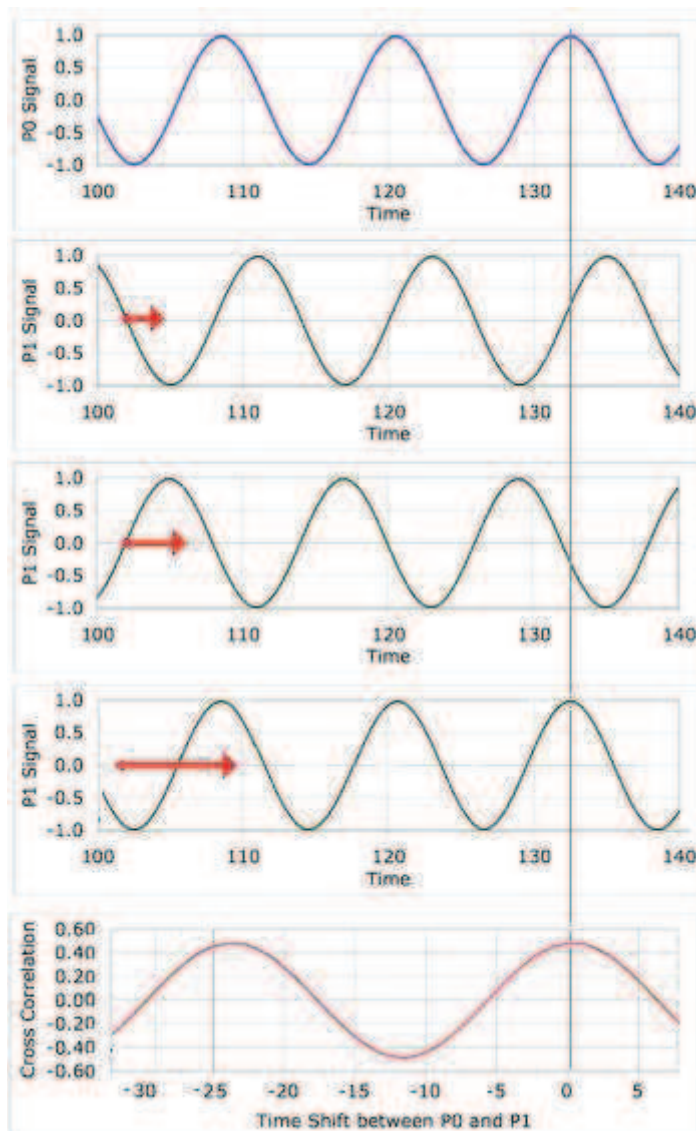


Figura 1-13 Correlación Cruzada de dos Señales Senoidales

El valor máximo de la correlación cruzada es independiente de algún tipo de acondicionamiento a la señal por ejemplo normalización, filtrado atenuación o

amplificación. Por ejemplo en la Figura 1-14 se muestran las variaciones de amplitud pero semejanzas en “secuencias” y fase.

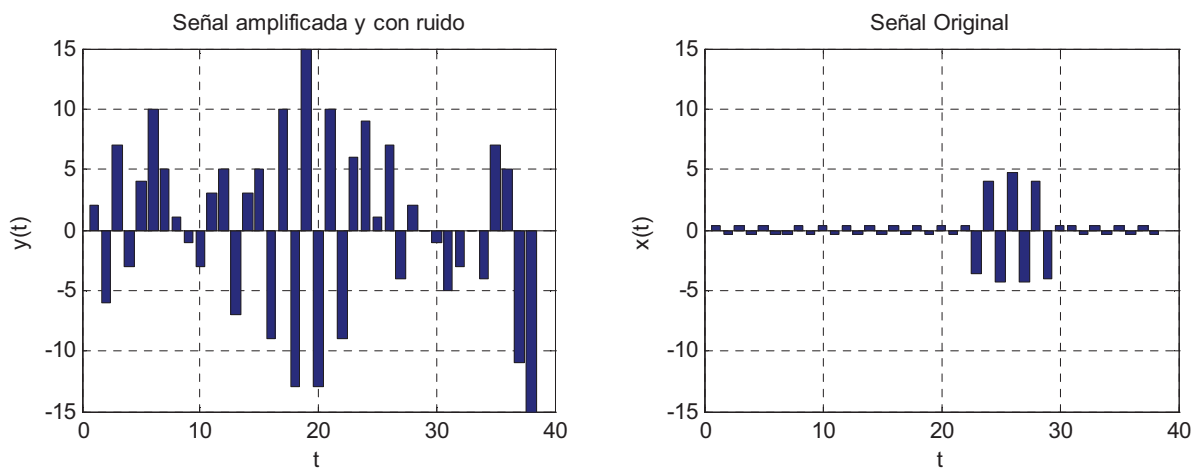


Figura 1-14 Señal con Ruido y sin Ruido

1.11. ENTORNO DE DESARROLLO DE INTERFAZ GRAFICA DEL USUARIO (GUIDE)

GUIDE (Graphical User Interfase Development Environment) es una facilidad dada por MATLAB, que permite crear Interfaces Gráficas para el Usuario GUI's (por su traducción del inglés Graphical User Interfaces), de manera que acelera el diseño y presentación de los controles de la interfaz, permitiendo programar cualquier trabajo de manera versátil al seleccionar, editar, arrastrar y personalizar propiedades. En su ejecución como producto final tiene el total soporte de las características matemáticas de MATLAB, incluyendo el servicio del control de los gráficos en MATLAB.

2. CAPÍTULO DOS. DISEÑO E IMPLEMENTACION DE SPOTID

2.1. INTRODUCCION¹³

En este capítulo se describe los factores que afectan al proyecto y sus requisitos, para luego desarrollar la implementación del software denominado SpotID. Se describe la creación de la bases de datos para la comparación de los spots publicitarios, detallando el despliegue de la información procesadas.

2.2. PERSPECTIVA DE SPOTID

Alrededor del mundo hay muchas compañías dedicadas a este trabajo y alegan tener software capaz de reconocer los anuncios publicitarios de manera rápida y certera. Los propietarios de dicho software, guardan celosamente las herramientas del “cómo lo hacen” y “cómo funciona”. Por lo que no es posible relacionar con algún software comercial, revisar su alcance o realizar comparaciones con el prototipo aquí desarrollado.

SpotID es dependiente de otros productos que puedan manejar hardware y otra sección que se encargará pre-procesar señales, por lo que, dependerá de software que permita:

- Grabar señales de TV y Radio.
- Extraer audio a señales de archivos de video.
- Editar archivos de audio.
- Presentar resultados, e;
- Imprimir detalles de resultados.

En el diagrama de bloques de la Figura 2-1 se puede ver la distribución del sistema de forma general.

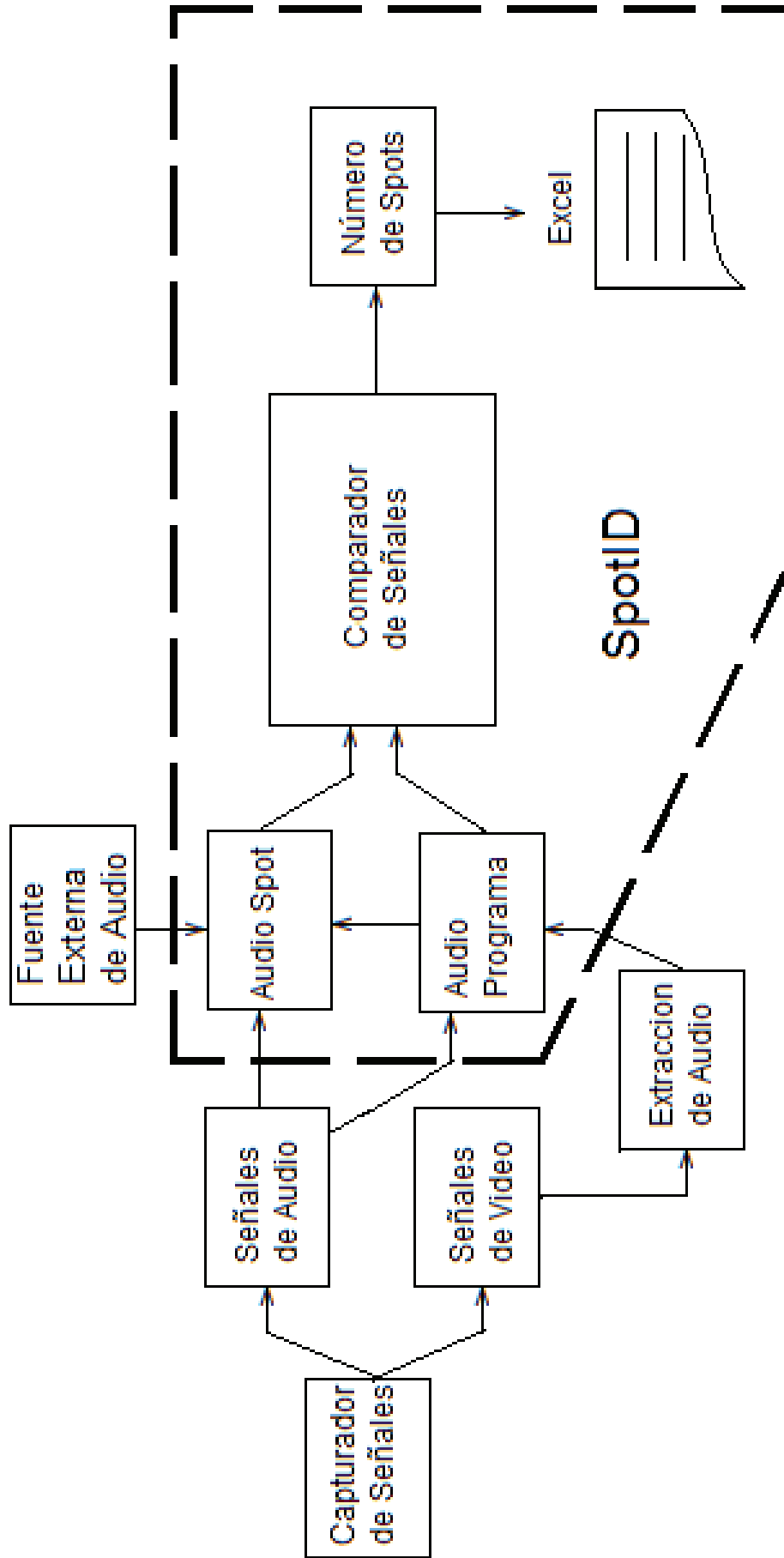


Figura 2-1 Diagrama de Bloques del Sistema SpotId

2.3.FUNCIONES DE SPOTID

La Figura 2-2 muestra el esquema de Funcionamiento del Sistema de reconocimiento de spots.

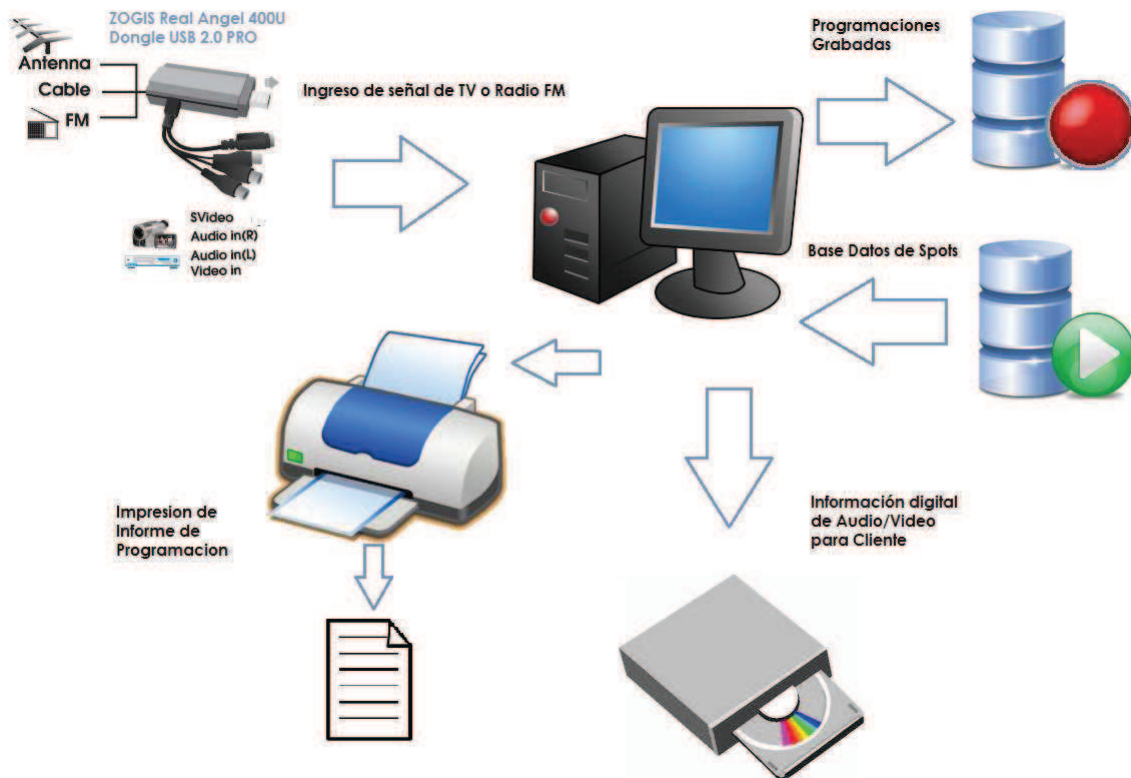


Figura 2-2 Esquema del Funcionamiento del Sistema

- La señal de Audio o Video es capturada e ingresada al computador puede ser grabada una estación por capturador a la vez.
- La grabación de audio y video de los programas radio o TV, se almacena en una base de de datos para luego ser procesadas
- Una segunda base de datos respalda el audio y video de spots que serán comparados con las programaciones grabadas.
- Terminada la grabación, con el software SpotID, se procede a cotejar los Audios de Programación y Spot en busca de las propagandas definidas por el operador.

- Luego del cotejamiento existe la opción de Imprimir el informe del cotejamiento y también realizar copias de las Programaciones grabadas y/o spots.

2.4. CARACTERISTICAS DE LOS USUARIOS

El software estará enfocado al uso de personas con conocimientos básicos en manejo de un computador de escritorio, impresora, capturador de video y manipulación digital de archivos de audio y video.

2.5. RESTRICCIONES

SpotID será implementado considerando las siguientes restricciones:

- Limitaciones de hardware
- Interfaces con otras aplicaciones
- Funciones de control
- Programación en Matlab
- Situaciones críticas de SpotID

2.5.1. LIMITACIONES DE HARDWARE

La elección de capturador de señales y los tipos de formatos que presente dependerá del software de programación. A la vez que se debe considerar el tamaño de los archivos a almacenar que al tratarse de video se requerirá de una eficiente capacidad de procesamiento y memoria.

La tarjeta capturadora permitirá grabar la señal, únicamente de una estación emisora de radio o televisión. En lo concerniente a la impresión de los resultados existe libertad de elección del hardware.

2.5.2. RELACION CON OTRAS APLICACIONES

La grabación, almacenamiento y conversión de audio y video están a cargo de tres aplicaciones externas a SpotID. Las cuales se adaptarán al tipo de formato de

audio y video que sea compatible con el software de programación. Todas estas aplicaciones podrán estar dentro del mismo computador que se ejecuta SpotID.

2.5.3. FUNCIONES DE CONTROL

Dentro de lo que será SpotID las subrutinas de edición de audio y video demandan mucha memoria por lo que se deberán realizar con aplicaciones externas a SpotID.

2.5.4. PROGRAMACIÓN EN MATLAB

Matlab exclusivamente puede leer y reproducir archivos con formatos de audio WAV y ACC, por lo que será imprescindible que los archivos de audio a grabar tengan este formato. Bajo las mismas condiciones trabaja la herramienta de GUIDE de Matlab.

2.5.5. SITUACIONES CRITICAS DE SPOTID

Existen dos puntos a considerar, uno de ellos es la calidad de señal mínima necesaria para que los resultados de la comparación estén libres de ambigüedades; y la segunda situación crítica es el tipo de codificación que permita invertir el mínimo de recursos de memoria.

2.6. SUPOSICIONES Y DEPENDENCIAS

SpotID dependerá del formato de los archivos de audio que la tarjeta capturadora de video y las aplicaciones de conversión de audio puedan presentar. La aceleración del procesamiento depende de los equipos en que se corra SpotID. Se incluye a esta última característica el sistema operativo en donde se hospedará SpotID.

2.7. REQUISITOS ESPECIFICOS

Aquí se presenta los requisitos para permitir diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y para luego realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos.

Se especifica los comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas.

Entonces, apoyados en las funciones de SpotID, presentadas en el Capítulo 2.3 y apoyados en la Figura 2-2 se describen los requisitos previos al diseño.

2.7.1. CAPTURA DE DATOS Y ALMACENAMIENTO

En el software de VivaTV, se establecerá la captura o grabación de la programación de cualquier canal de televisión a en:

Formato Video	Resolución	Velocidad de bits	
MPG	720x480p	2224kbps	

Formato Audio	Velocidad de bits	Frecuencia de muestreo	
MP3	224kbps	48khz	

VivaTv tiene una nomenclatura establecida para nombrar los archivos. La estructura es de la siguiente manera:

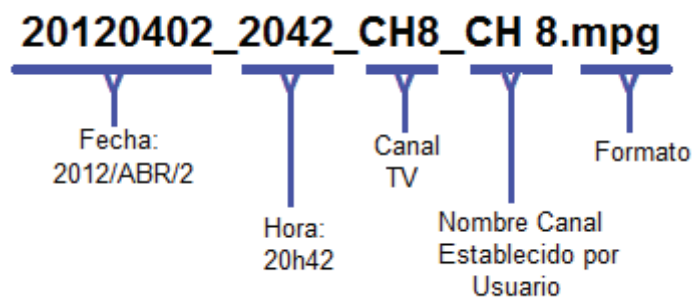


Figura 2-3 Nomenclatura de Archivo Programación

A partir de esta estructura, el programa a diseñarse, podrá procesar la siguiente información:

- Fecha de Grabación
- Hora de Grabación
- Duración de la Grabación.
- Canal y Nombre de Estación
- Tipo de Programación (A, AA, AAA)



Figura 2-4 Ejemplo de Archivos Programación.

La captura no se excederá de las tres horas continuas de grabación, con el fin de que no se vea afectado el rendimiento del computador por el hecho de manejar archivos demasiado grandes. Aproximadamente un archivo de 3 horas de duración requerirá de 3Gb de memoria.

Cabe recalcar que se debe ajustar la antena en busca de buena calidad de señal, que está a criterio de la percepción visual y auditiva del operador. Aunque, consimejor considerar técnicamente como buena la relación Señal a Ruido de 53dB¹⁴

2.7.2. PRE-ACONDICIONAMIENTO DE AUDIO Y VIDEO

Un paso previo al cotejamiento es la separación del audio del video, puesto que el programa procesa exclusivamente dos señales de audio en formato WAV. Estas son: el Audio de la Programación de duración de decenas de minutos; y, otra de menor longitud el Audio de Spot, menos de 45 segundos.

2.7.2.1. Audio de Programación (AP)

Luego de capturar el video, en este caso a un programa de X estación televisiva, le será extraído el audio con Free Video to MP3 Converter.

En este software se establece como salida un archivo de AP en formato WAV de 8khz de frecuencia de muestreo, con calidad monofónica. Este tipo de frecuencia de muestreo facilita la manipulación en Matlab en lo que respecta al tamaño del archivo.

Otra razón para no exigir mayor “calidad” de audio, es que, en el cotejamiento las operaciones computacionales utilizadas en la correlación, no requiere de mayor fidelidad ni filtros de ruidos, recalcando nuevamente el criterio del operador en el ajuste de la antena.

2.7.2.2. Audio de Spot (AS)

Este archivo puede ser extraído del Audio de Programación como también de un archivo independiente generado de una fuente externa que registre un audio de cualquier spot publicitario, pero cumpliendo con características de formato WAV y frecuencia de muestreo 8khz.

2.7.3. COTEJAMIENTO

Una vez que se tiene los audios de Programación y Spot en formato WAV, se procederá a usar el SpotID para el cotejamiento.

El programa SpotID está diseñado para procesar archivos de audio, por lo que su uso sólo se realizará luego de pre procesar archivos de video para la extracción de su audio. La extracción del audio de archivos de video se lo realizará mediante el software DVDVideoSoft Free Studio.

El software presenta una ventana principal, en la cual es visible los botones activos para realizar el proceso inicial. Básicamente consta de cuatro pasos:

- Selección del Archivo de Audio Programación.
- Selección del archivo de Audio Spot,
- Cotejamiento
- Resultados

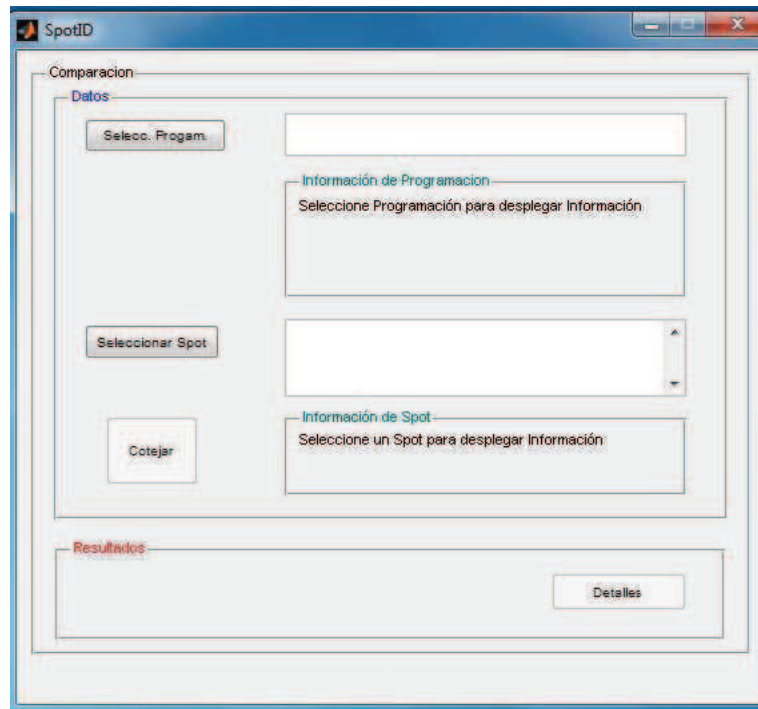


Figura 2-5 Ventana Principal de SpotID

2.7.3.1. Selección del Archivo de Audio Programación.

Una vez elegido el Audio Programación, se despliega información del archivo.

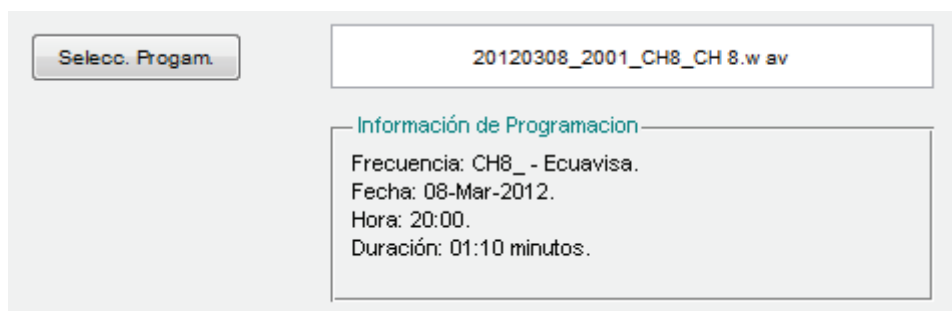


Figura 2-6 Selección de Archivo Programación

2.7.3.2. Selección del archivo de Audio Spot,

De similar procedimiento al anterior, una vez cargado el Audio Spot, se genera información del spot.

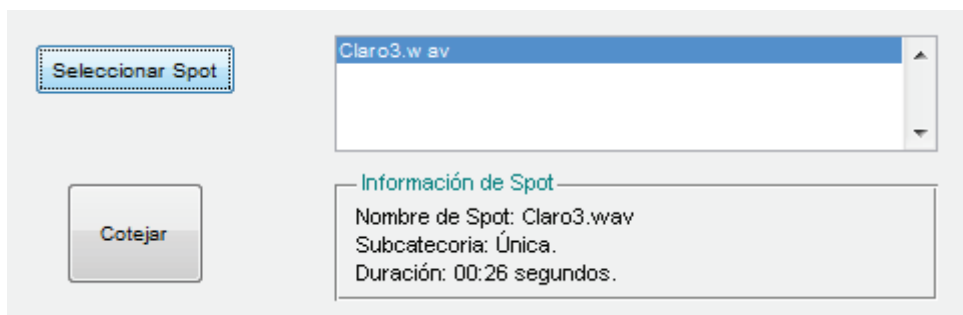


Figura 2-7 Selección de Archivo Spot

2.7.3.3. Cotejamiento

Seleccionados los dos audios, se procede a dar clic en el botón Cotejar. Luego de lo cual aparece en pantalla una barra de avance del proceso. Una vez terminado se podrá optar por ver el informe.

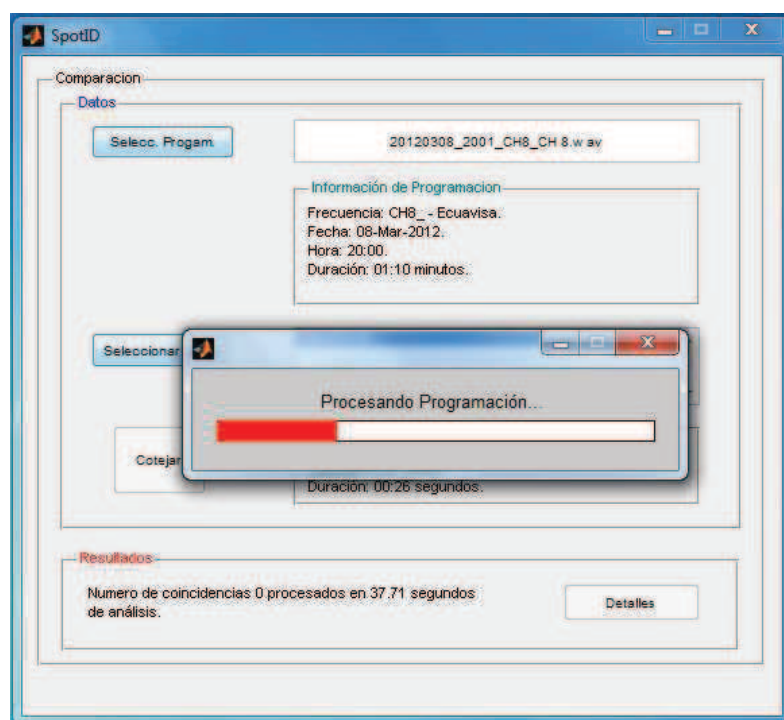


Figura 2-8 Momento de Procesamiento de Audios Programación y Spot

2.7.3.4. Resultados

Los resultados del cotejamiento se observa pie de la ventana. Al dar click en el botón Detalles, se despliega una segunda ventana mostrando los tiempos en que se hallaron semejanza, si existen, del spot en el Audio Programación.

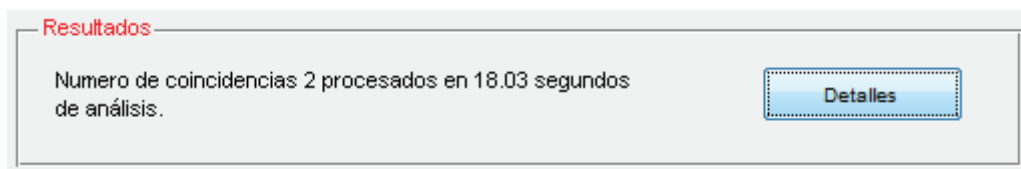


Figura 2-9 Pie de Ventana en la que se muestra brevemente el resultado

2.7.3.4.1. Reporte

En la ventana de Resultados, Figura 2-10, se puede acceder al reporte de resultados en formato Excel para poder visualizarlo y también para imprimir una hoja como se muestra en la Figura 2-11.

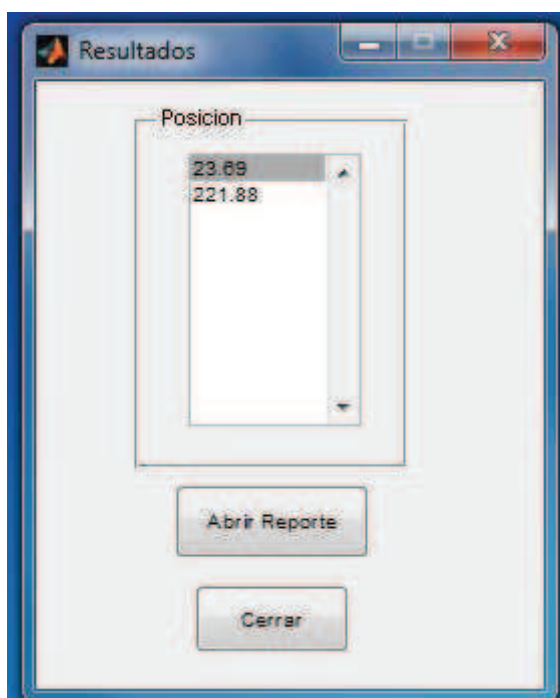


Figura 2-10 Ventana luego de presionar botón Detalles de la Ventana Principal

2.1. EFICIENCIA

SpotId debe mantener una aceptable rapidez en el procesamiento de datos, para que de esta manera pueda reemplazar la labor de un operador humano realizando la misma tarea.

2.2.PRECISIÓN

El prototipo debe reducir en lo mínimo posible los falsos hallazgos de spots, para de esta manera reducir la supervisión humana

2.3. IMPLEMENTACIÓN DEL SOFTWARE SPOTID

El desarrollo de SpotID está realizado en lenguaje MATLAB versión 7.4.0.287 en un computador HP dv7-3065dx con Sistema Operativo Windows 7 Premium 64-bit y Procesador AMD Turion II Ultra M600 Caspian 45nm Technology.

La secuencia de implementación sigue como se estructura de la siguiente manera:



Figura 2-12 Secuencia de Procesamiento General

2.3.1. SELECCIÓN DE AUDIOS A COTEJAR

Dentro del primer paso, se procede a programar para que dentro de SpodID sea posible seleccionar archivos con formato WAV. Específicamente se escogen dos archivos:

- Audio Programación; y,
- Audio Spot.

2.3.1.1. Selección de Audio Programación

Para abrir los archivos de Audio Programación se procede con la siguiente estructura:



Figura 2-13 Secuencia de Procesamiento de Audio Programación

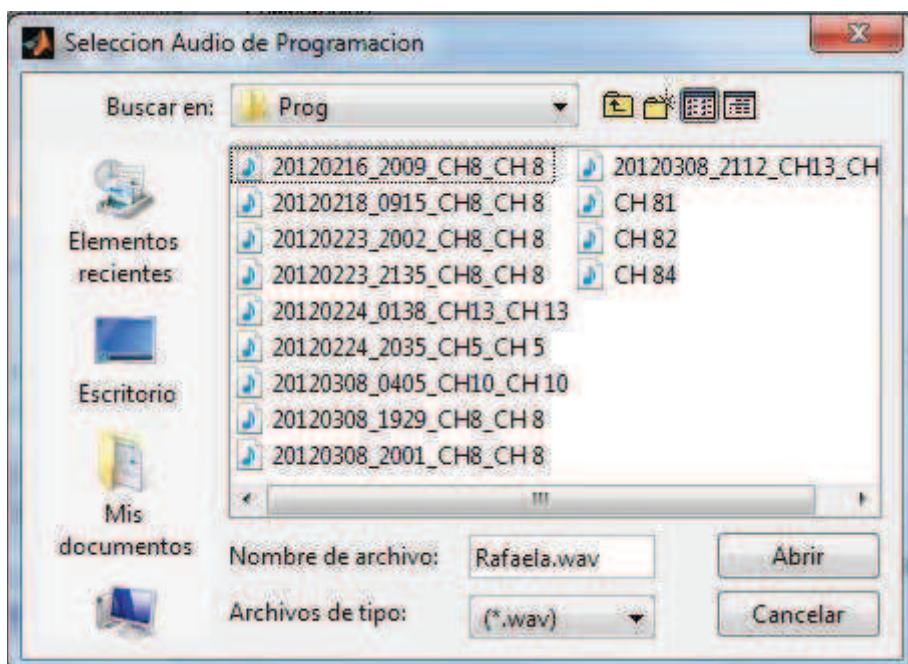


Figura 2-14 Ventana de Selección de Audio Programación

Se escriben líneas de código con el fin de que se pueda usar las ventanas de apertura de archivos en ambiente Microsoft Windows, a ello se suma el filtro para seleccionar únicamente de archivos WAV.

2.3.1.1.1. Acondicionamiento de Audio Programación

En este punto de la implementación se tiene una restricción en el procesamiento de archivos en MATLAB, específicamente con la función *xcorr*, que es la que correlacionará archivos de audio.

Los archivos que intervendrán en el cotejamiento deben ser de máximo 2^{20} bytes, o sea, deben ser menores a 1.048MB. Lo que traducido en tiempo de grabación a 64kbps, son exactamente a:

$$\text{Tamaño Máximo} = 2^{20} \text{ Bytes} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ segundo}}{64 \text{ k bits}}$$

$$\text{Tamaño Máximo} = 1048576 \text{ bits} * \frac{8 \text{ bits}}{1 \text{ Byte}} * \frac{1 \text{ segundo}}{64000 \text{ bits}}$$

$$= 131.072 \text{ segundos}$$

Lo que equivale a 2 minutos con 10 segundos, aproximadamente.

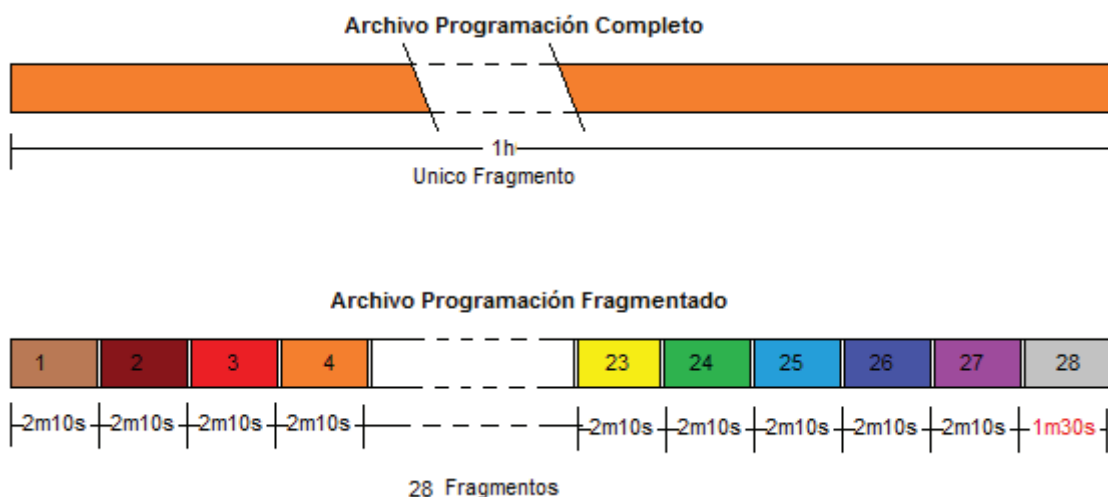


Figura 2-15 Fragmentación de Archivo Programación

Cómo la duración de los Audios de Programación va más allá de los 2m10s, entonces se procede a “fragmentar” en varios archivos con la longitud de tiempo requerida.

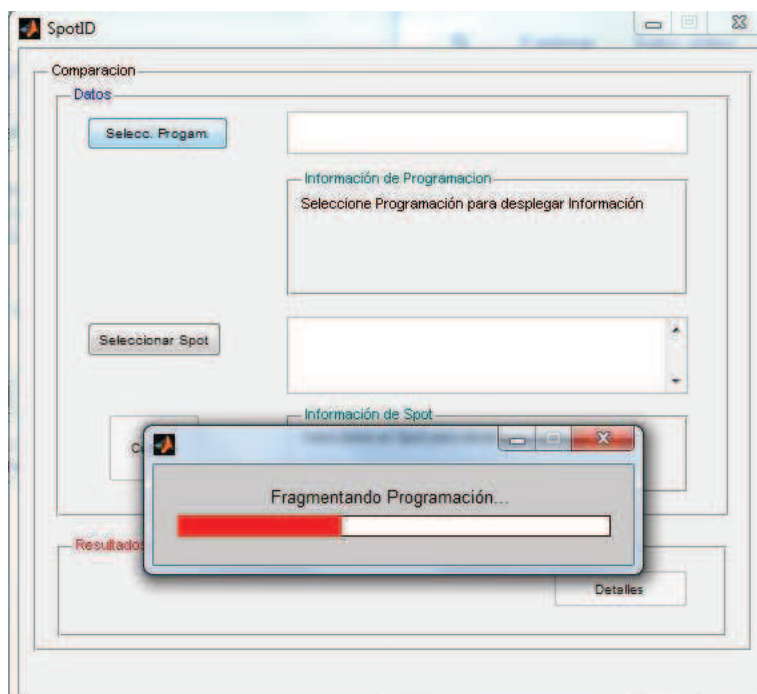


Figura 2-16 Ventana de Proceso de Fragmentación

Estos fragmentos se almacenan en una matriz de Programación Fragmentada. El número de Fragmentos dependerá del total de segundos que Archivo Programación dividido para 131.072 segundos. A continuación está un ejemplo de fragmentación de un Archivo Programación de una hora de duración.

$$\text{Número de Fragmentos} = 1\text{hora} * \frac{3600\text{segundos}}{1\text{hora}} * \frac{1\text{fragmento}}{131.072\text{segundos}}$$

$$\text{Número de Fragmentos} = 27.46\text{ fragmentos}$$

El archivo de una hora se divide en 28 fragmentos:

27 fragmentos	2min 10 seg
1 fragmento	1 min 30 seg (aprox)
28 fragmentos	1 hora (aprox)

Este proceso se realiza cada vez que se carga un archivo de Audio Programación. El tiempo de fragmentación depende del tamaño del archivo, en promedio en una Programación de una hora de duración, el proceso tarda unos 8 segundos.

2.3.1.1.2. Extracción de características Informativas.

Inmediatamente después de fragmentar, SpotID lee el nombre del archivo, asignado por el capturador Zogis RU400. Con el fin de obtener información del archivo que se decodifica para presentarla en la ventana respectiva.

Nombre de Audio Programación	Información Extraída del Nombre
<i>20120308_2001_CH8_CH 8.wav</i>	Frecuencia: CH8_ - Ecuavisa. Fecha: 08-Mar-2012. Hora: 20:00. Duración: 01:10 minutos.

Tabla 2.1 Decodificación del Nombre de Archivo de Audio Programación.

El nombre de la estación, esta almacenada en una base de datos dentro de una subrutina que nombrará e imprimirá en el informe el archivo. Además se agrega la clasificación de audiencia del programa.

Horario del Programa	Clasificación
23h00 a 16h59	A
17h00 a 18h59	AA
19h00 a 22h59	AAA

Tabla 2.2 Clasificación de Audiencia del Programa.

2.3.1.2. Selección de Audio Spot

Estos archivos también se cargarán de manera similar a los archivos de Programación.

Nomenclatura	Ejemplo:
Nombre de Anunciante – Tipo – Categoría	Claro – 2x1 – Telefonía.wav

Tabla 2.3 Decodificación de Nombre de Archivo de Audio Spot

Aquí no existirá la fragmentación, debido a que los archivos de Audio Spot se establece que no pasarán de los 45 segundos.

Luego en la extracción de características existe una diferencia al Audio Programación, pues el nombre del archivo Spot está establecido por el operador, pero deberá cumplir con la nomenclatura de la Tabla 2.3.

El momento de nombrar es configurado al momento de ingresarlo a la base de datos de Audios Spot. Esta nomenclatura permitirá desplegar información del archivo Audio Spot que se correlacionará con el Audio Programación.

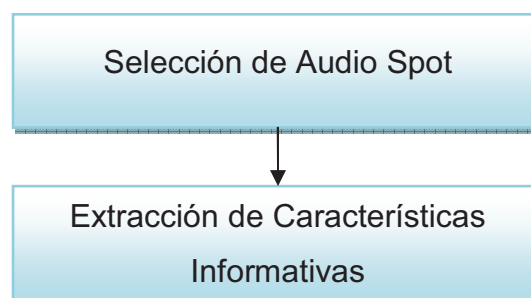


Figura 2-17 Secuencia de Selección de Archivo Spot

La información de Programación y Spot servirán para completar el Reporte de Cotejamiento al final de todos los procesos

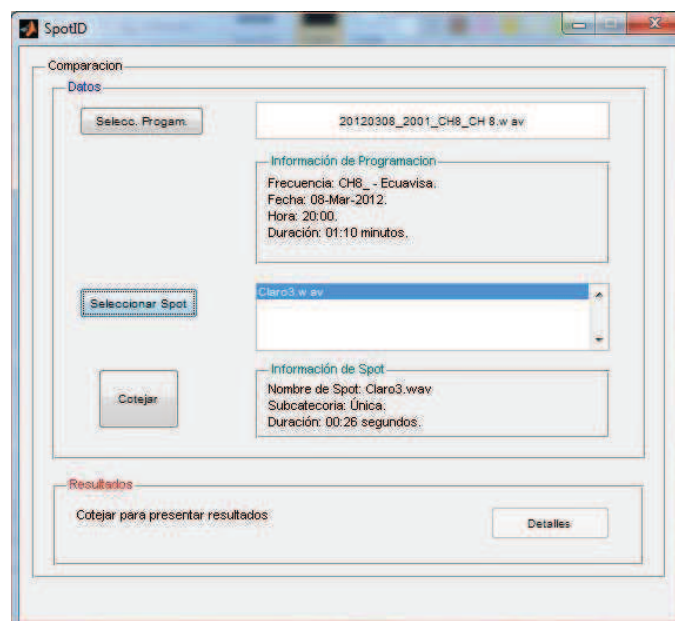


Figura 2-18 Ventana con Audios Cargados, listos para ser Cotejados

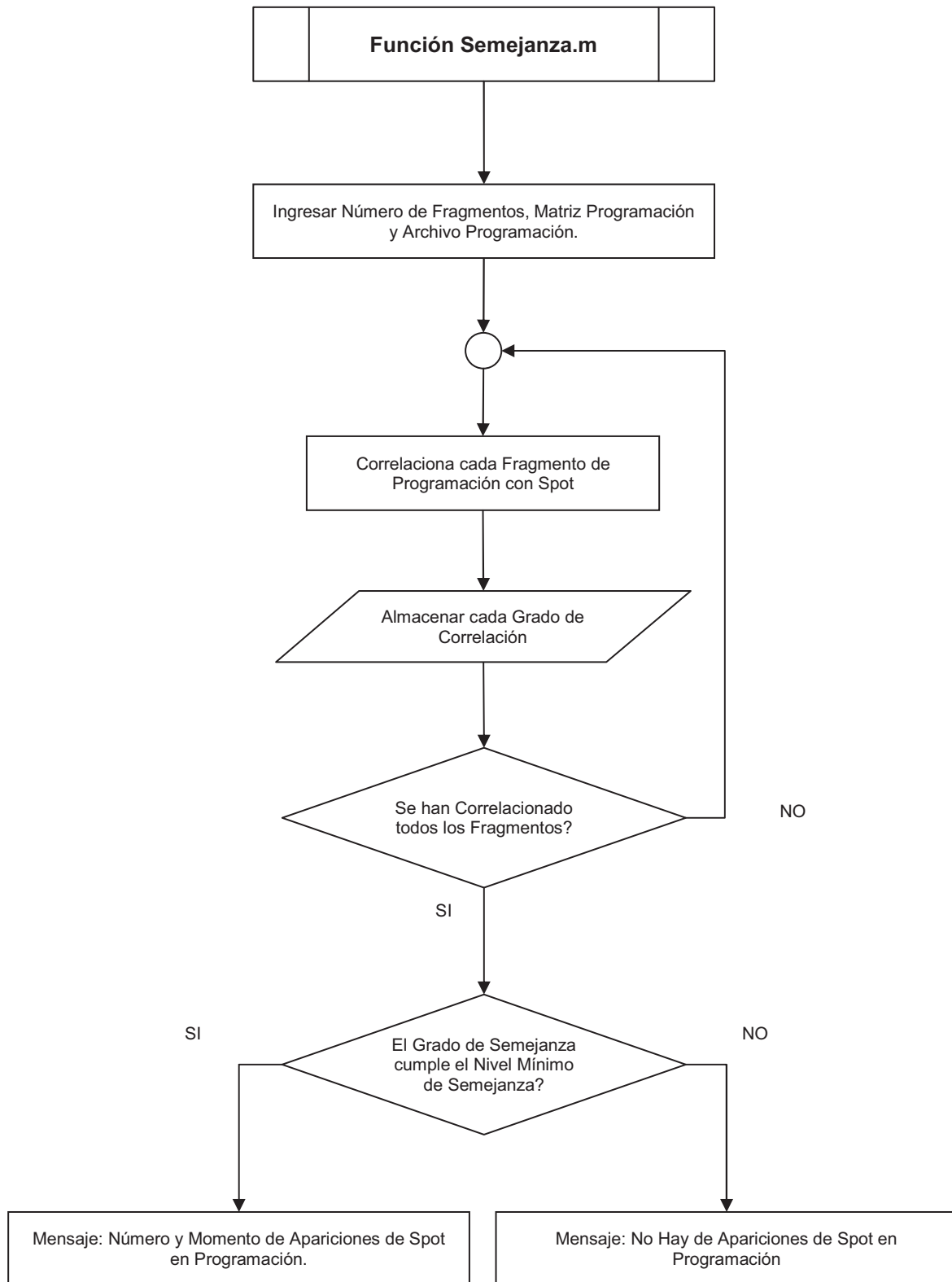


Figura 2-19 Flujograma de determinación del Nivel de Similitud

2.3.2. CORRELACIÓN DE AUDIOS SELECCIONADOS

Una vez cargado el Audio Programación y Audio Spot se procede a correlacionar, que ocurre una vez presionado el botón Cotejar. Esta acción invoca a ejecutar una función llamada semejanza.m

Esta función realiza los pasos de flujograma visto en la Figura 2-19

Además, de los archivos de audio, se necesita el número de fragmentos en que se ha dividido el Archivo Programación, este dato permite conocer el número exacto de correlaciones que deba hacer entre la matriz de Audio Programación frente al Audio Spot.

El resultado del cotejamiento, es un número que manifiesta el Grado de Semejanza entre dos archivos.

2.3.2.1. Grado de Semejanza

Debido a que las muestras de audio tiene valores de amplitud negativos y positivos, específicamente entre -1 y 1, el resultado de la correlación puede ser un número entero que va entre $-\infty$ y $+\infty$, que, para ser manejable se aproxima a un número absoluto. Este resultado, de valor absoluto, se lo denominará Grado de Correlación.

Por pruebas experimentales el Grado de Semejanza tiene valores entre 0 y 7000. Aunque no exista similitud en dos archivos, siempre habrá un valor máximo de semejanza. Para establecer qué número máximo reflejará una similitud, se ha establecido un Nivel Mínimo de Semejanza.

2.3.2.2. Nivel Mínimo de Semejanza

El Nivel Mínimo de Semejanza asegura, que, el grado de similitud entre las dos muestras es muy probable. Este límite numérico se ha determinado mediante

pruebas, está por encima de los 1500. En el Capítulo 3 se demuestra porqué existe un Nivel Mínimo de Semejanza.

Se toma como ejemplo la correlación los audios de la Figura 2-20; entonces en la Figura 2-21 representaría el resultado de esa correlación, la cual muestra un Grado de Semejanza Máxima de 440, aproximadamente a los 124 segundos. Lo que indicaría que al estar por debajo del Nivel Mínimo de Semejanza estas señales tienen una similitud pobre.

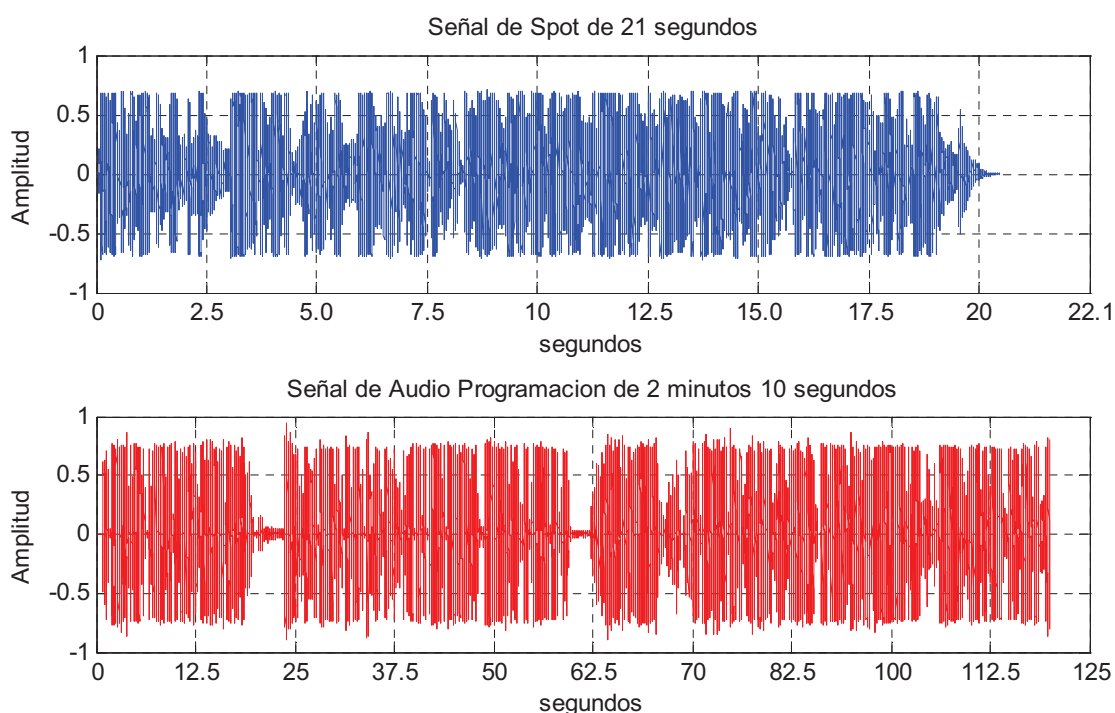


Figura 2-20 Representación de Señales Audios en Amplitud VS Tiempo

Tomando un segundo ejemplo, si se realizara el cotejamiento de dos señales iguales, es como si se estuviera haciendo la correlación de la misma señal; esto se denominaría una Auto Correlación o Auto Semejanza . La Figura 2-22 muestra el resultado de una Auto Semejanza y permite aclarar el panorama del Nivel Mínimo de Semejanza.

Como se han cotejado dos señales idénticas, entonces refleja que el Grado de Semejanza es un poco mayor a 12000.

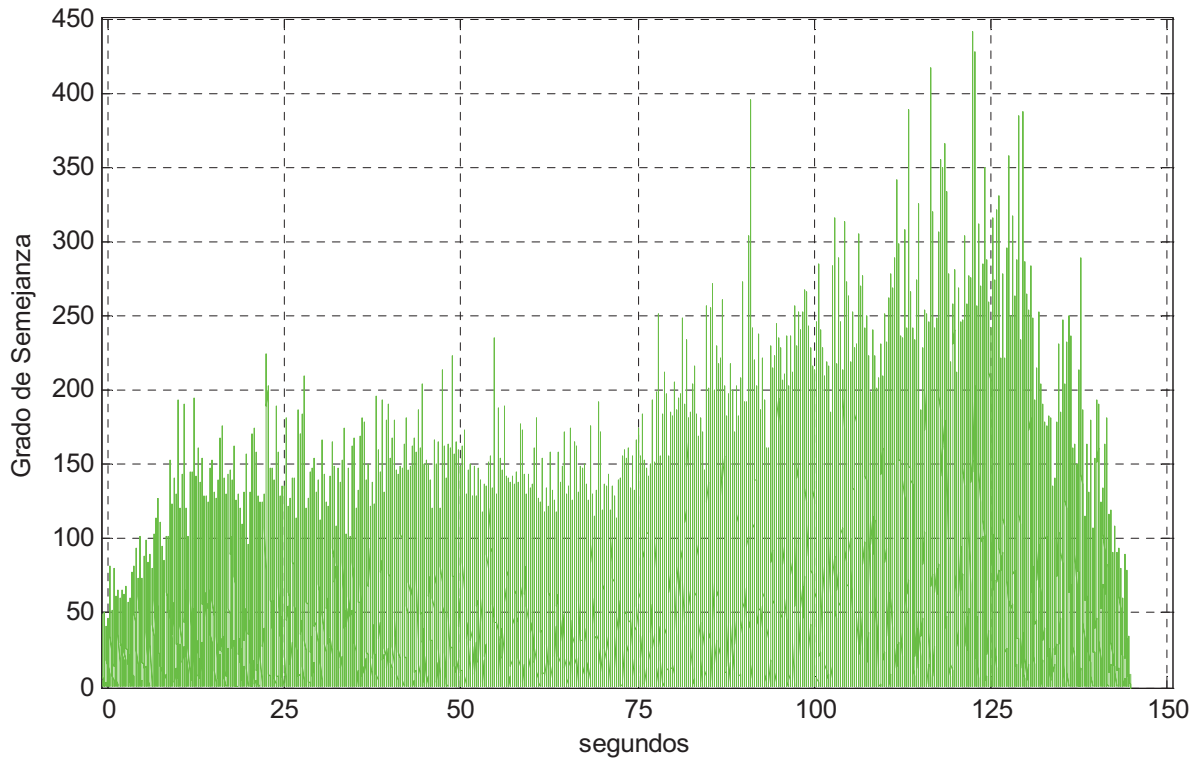


Figura 2-21 Representación de Similitud en Amplitud y Tiempo

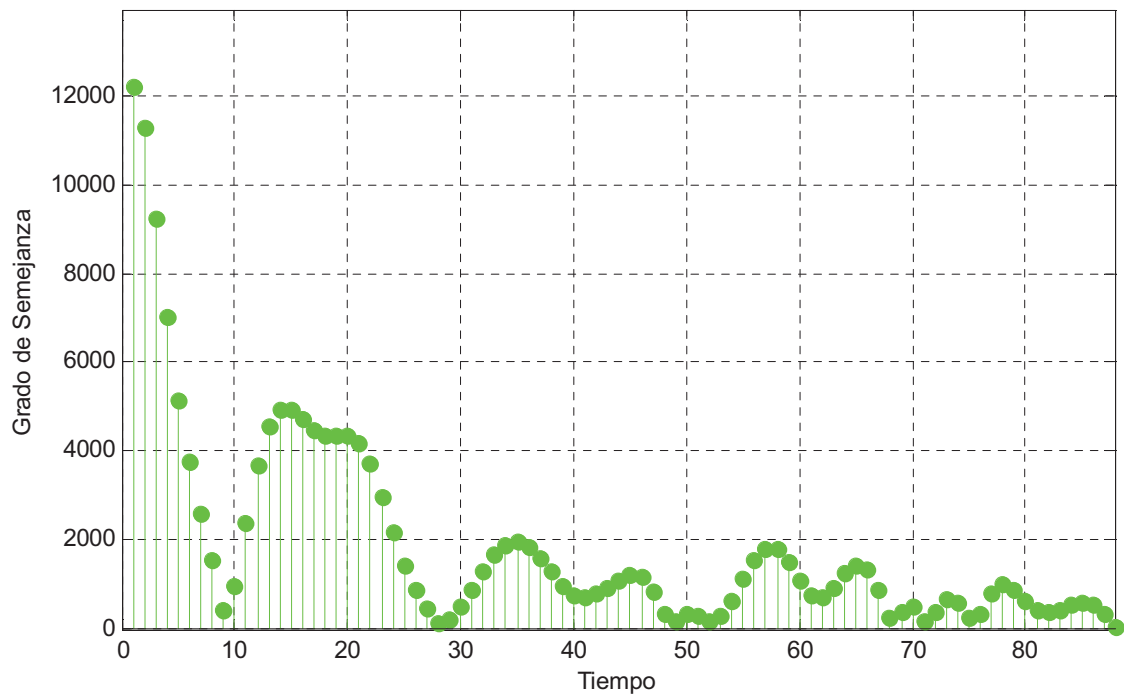


Figura 2-22 Representación de Auto Semejanza

Al cotejar otras dos señales idénticas se tiene el resultado mostrado en la Figura 2-23. En esta ocasión el grado de correlación máxima llega hasta casi a los 20000.

La variación del valor máximo de una autosemejanza y otra dependerá, exclusivamente, de las características de los audios Auto Ccorrelacionados.

Así, en la correlación de los Audios Programación y Spot, como la Figura 2-21, Lo que se toma en cuenta, que, el Grado de Semejanza sobrepase el Valor Mínimo requerido para sentenciar si existe o no una similitud entre dos señales.

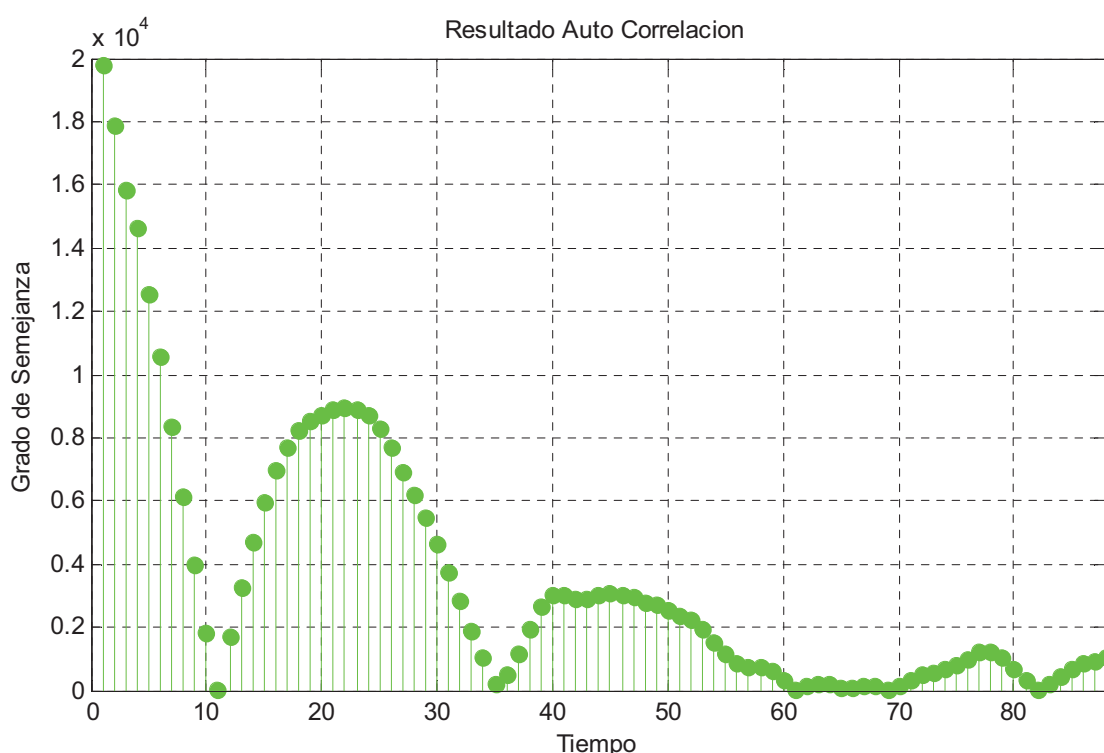


Figura 2-23 Auto Correlación de Señales

2.3.3. MOSTRAR RESULTADOS

Los resultados del cotejamiento o correlación son tratados para darles formatos que permitan ser entendibles al conocimiento del operador. Además de reflejar en pantalla estos resultados se provee en una hoja imprimible como muestra la Figura 2-11 del Reporte de Escaneo.

Los datos generados del procesamiento son almacenados en un archivo de datos denominado *reporte.xls*. Posteriormente se puede obtener la opción de imprimir un reporte.

El reporte imprimible se facilita por medio del programa Microsoft Excel 2007. Matlab permite la interacción con Microsoft Excel 2007, gracias a funciones como: *xlsread* y *xlswrite*. Estas funciones permiten abrir y editar, respectivamente, archivos con extensión xls y xlsx.

3. CAPÍTULO TRES. PRUEBAS Y RESULTADOS DEL SOFTWARE IMPLEMENTADO.

3.1. INTRODUCCION

En este capítulo se detalla los resultados del trabajo para comprobar la validez del software desarrollado. Tal como se explica en el Capítulo 1.2, uno de los objetivos del prototipo SpotID, es la optimización del tiempo, de tal manera que las pruebas apuntarán a dar mayor connotación a este particular. A su vez analiza certeza de hallar las apariciones publicitarias sin reportar “falsos positivos” (error de SpotID al notificar como una aparición de spot cuando no lo es).

Las pruebas se realizan con tres Archivos de Programación de una hora de duración, aproximadamente; que fueron grabadas bajo circunstancias normales de recepción. En estos archivos se escanearon 30 Spots, cada spot tiene diferente duración así como diferente calidad de audio. Además, se realiza un detalle de costos de la implementación del software en su desarrollo, construcción y aplicación.

3.2. RESULTADOS DEL SOFTWARE SPOTID

3.2.1. CAPTURA DE VIDEOS Y AUDIOS

La recepción de la señal es recogida con una antena de aire, la cual estará conectada directamente al capturador ZOGIS Real Angel 400U Dongle USB 2.0 PRO. Este hardware está acoplado al computador mediante puerto USB. La alimentación eléctrica del capturador es de 5V, la misma que la realiza a travez del puerto USB.

Con esta configuración de hardware únicamente puede ser grabada una estación a la vez. El computador podrá reproducir y grabar una señal radial o televisiva al ejecutar el software VivaTV, suministrado en el paquete capturador.

Con el software VivaTV, se configura la captura del video o el audio, con el formato de la Tabla 3.1 y Tabla 3.2

Formato Video	Resolución	Velocidad de bits
MPG	720x480p	2224kbps

Tabla 3.1

Formato Audio	Velocidad de bits	Frecuencia de muestreo
MP3	224kbps	48khz

Tabla 3.2

Con estas configuraciones, un video o audio grabado de alrededor una hora de duración, tendrán un tamaño reflejado en la .Tabla 3.3

Formato	Duración	Tamaño
Video.mpeg	1 hora	1 GB (aproxim)
Audio.mp3	1 hora	56 MB (aproxim)

Tabla 3.3

Estos tipos de formatos permiten tener la calidad suficiente para:

- Almacenar
- Duplicar; y,
- Procesar

El almacenamiento lo realiza directamente el programa VivaTV, con los formatos y nomenclatura descrita en el Capítulo 2.3.1.

El duplicado, en memorias flash u óptica, es realizado en el ambiente Windows 7 de Microsoft, con la intención de realizar un respaldo de la información o a su vez la entrega de pruebas del proceso.

Luego del almacenamiento y/o duplicado, el archivo capturado empieza el procesamiento con la extracción del Audio, en caso de que sea video; si el archivo es un audio, se pasa directamente al cotejamiento.

3.2.1.1. Estaciones de Radio y Televisión

Para realizar las pruebas se las realiza en medios con cobertura en la ciudad de Quito, y se escogen las siguientes estaciones:

Canal	Estación	Programa	Tipo de programa	Hora de Grabación	Duración
4	Teleamazonas	24 Horas	Noticiero	20h00	1 hora
8	Ecuavisa	Cuchicheos	Novela	20h45	1 hora
12	Canal Uno	En Carne Propia	Reportajes	21h00	1 hora

Tabla 3.4 Estaciones Televisivas

Frecuencia	Estación	Programa	Tipo de programa	Hora de Grabación	Duración
92.5	Exa FM	La Papaya	Concursos	10h00	2 horas
90.5	Disney	El Expreso	Canciones	15h00	2 horas
106.5	Canela	Radiación Temprana	Talk Show	8h00	2 horas

Tabla 3.5. Estaciones de Radio FM.

El criterio para escoger las estaciones de Radio y Televisión está basado en “rating” (nivel de audiencia), pues tienen la mayor variedad de apariciones publicitarias debido al mismo hecho de tener mayor audiencia.

3.2.2. EXTRACCION DE AUDIO

Este proceso exclusivamente se lo hará para los Archivos de Video con la ejecución del software Free Video to MP3 Converter. Figura 3-2. y se lo utilizará para obtener el audio de Programación y Spot.



Figura 3-1 Extracción de Audio

3.2.2.1. Audio Programación

A este programa se ingresa archivos de video con el formato explicado en la sección 3.2.1 de este Capítulo. El formato de salida es un archivo únicamente de audio en formato WAV detallado en la Tabla 3.6 Formato de Salida

Formato	Canales	Velocidad de bits
WAV	Mono canal	64kbps

Tabla 3.6 Formato de Salida

El nombre del archivo de audio de salida mantendrá la misma nomenclatura que el archivo de video de entrada. Así mismo mantendrá el tiempo de duración. El tamaño es mucho menor, como se puede comparar en la Tabla 3.7 con la conversión de varios archivos de diferente tiempo de duración.

Duración	Tamaño Video	Tamaño Audio
3h13m56s	3.06GB	177MB
1h33m17s	1.48GB	85.4MB
1h10m33s	1.11GB	64.5MB
1h02m08s	981MB	56.8MB
31m26s	508MB	28.7MB

Tabla 3.7 Tamaño de Archivo Audio y Video

Esta disminución del tamaño, debida a la conversión, beneficia en la disminución del tiempo de procesamiento. Debido a que, al momento de cotejar, el computador cargará y procesará menos datos a su memoria y procesador, respectivamente.

El tiempo de conversión empleado por Free Video to MP3 Converter, dependerá de la duración del video. En la Tabla 3.8 muestra los tiempos de conversión con diferentes archivos.

Tiempo de Duración del Video	Tiempo de Conversión
3h13m56s	2m4s
1h33m17s	58.6s

1h10m33s	42.2s
1h02m08s	37.4s
31m26s	18.1s

Tabla 3.8 Tiempos de Conversión

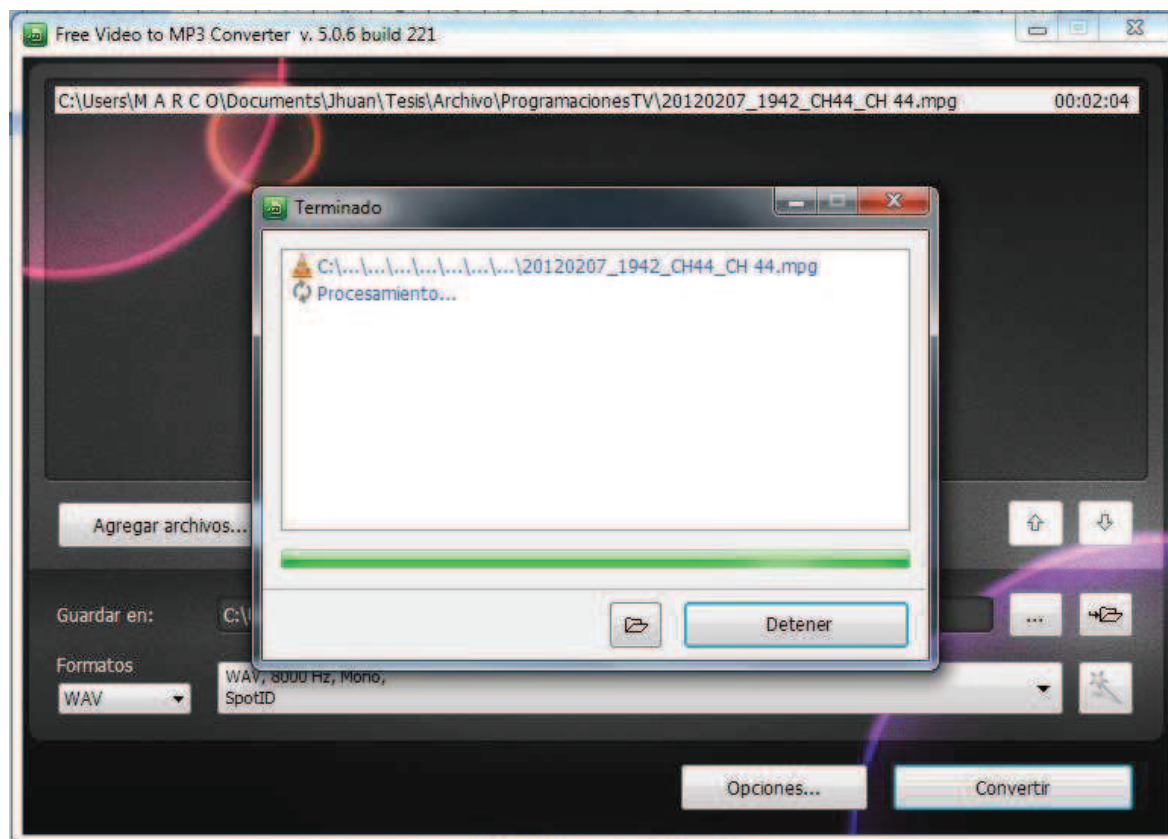


Figura 3-2 Software en Conversión Video a Audio

3.2.2.2. Audio Spot

El Audio Spot es obtenido del Audio Programación. Puesto que dentro de este último se asume que contiene varios anuncios publicitario, se abre el archivo en el programa Audacity® 1.3.13-beta.

En la ventana principal de Audacity, Figura 3-3, se muestra Audio Programación. Manualmente se busca y elige la fracción que contenga una propaganda, para luego copiarla.

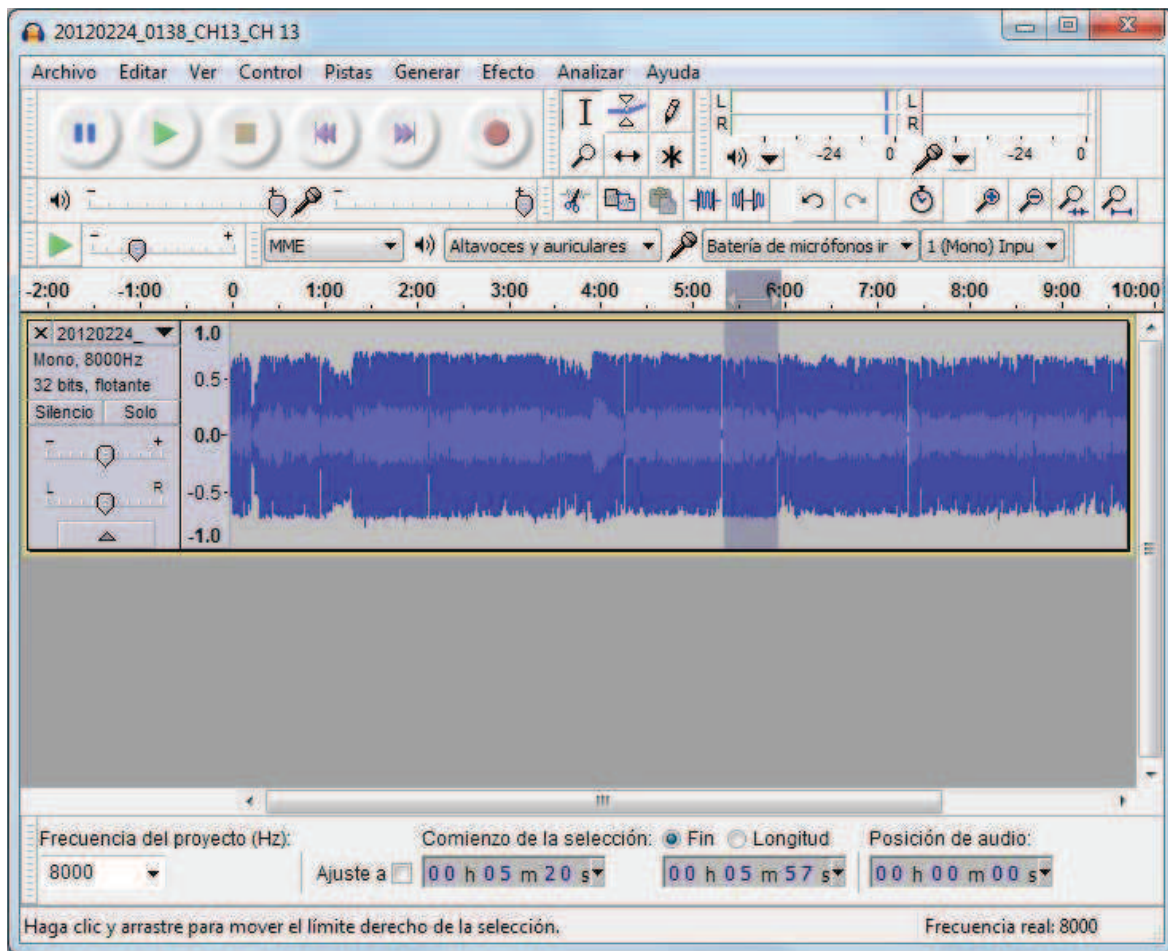


Figura 3-3 Selección de Audio Spot

El fragmento copiado es guardado en un archivo nuevo con el nombre respectivo respetando el formato de nombramiento visto en el Capítulo 2.4.1.2. como puede verse resaltado en rojo en la Figura 3-4 la longitud del nuevo audio no excede de los 18 segundos.

Así se pueden obtener varias muestras de publicidad, o incluso frases de una propaganda o la cortina musical que no es más que la música que se intercala para marcar las diferentes partes de un programa televisivo sea este de Concurso, Variedad, Noticiero o Novela.

El tiempo empleado en la obtención de “un” solo Audio Spot está sujeto a la destreza del operador para realizar los pasos antes mencionados. Al realizar un banco de datos de los Audios Spot, se tiene un tiempo promedio de 1 minuto y medio.

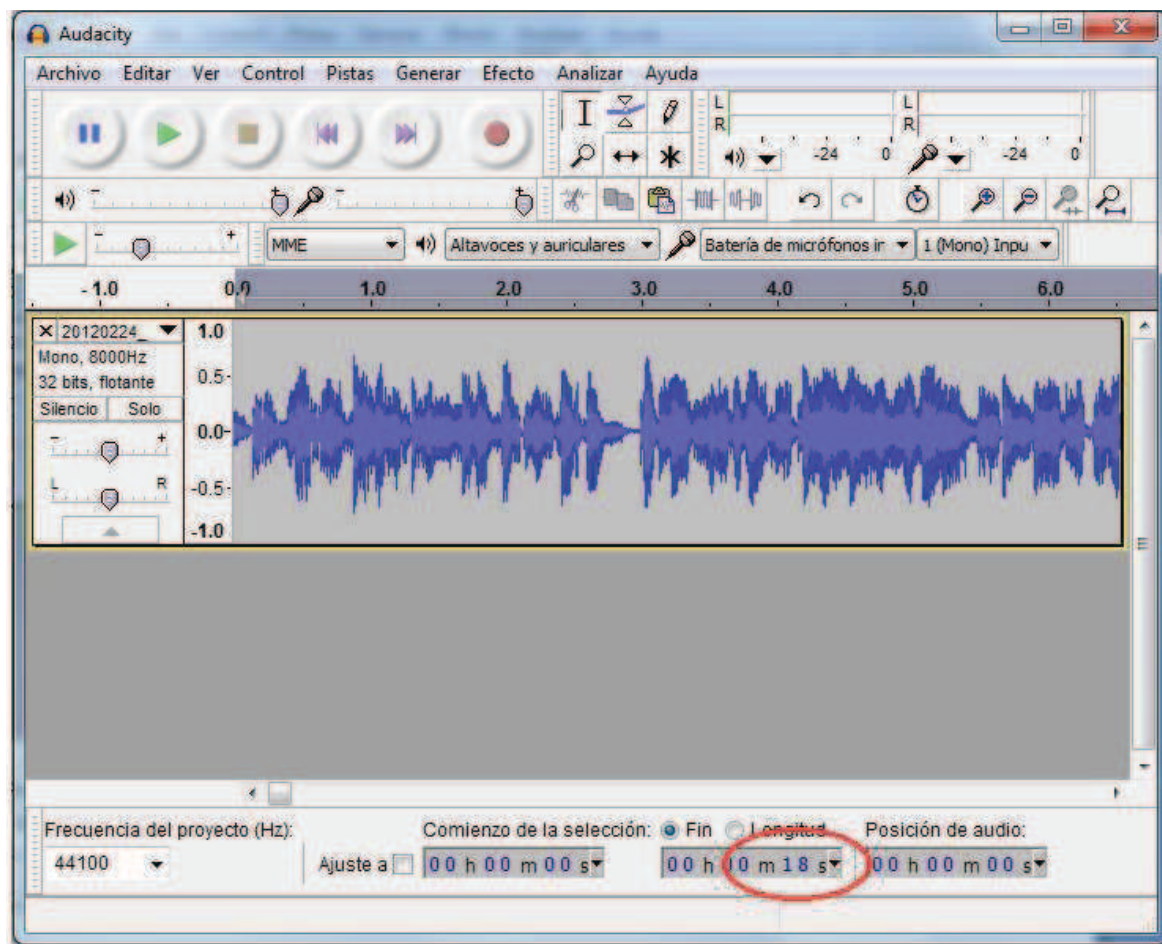


Figura 3-4 Archivo Spot

3.2.3. COTEJAMIENTO DE SEÑALES DE AUDIOS

Previo al Cotejamiento, se procede a cargar los dos archivos a correlacionar. Debido a que los Archivos de Programación deben fragmentarse al cargar, se debe analizar el tiempo empleado en esta tarea.

El archivo Audio Spot carga inmediatamente sin que exista necesidad de analizar su tiempo de carga.

3.2.3.1. Carga y Fragmentación de Audio Programación

Para las pruebas de carga y fragmentación se procede a cargar archivos de características detallada en la Tabla 3.9

Duración	Tamaño Audio
3h13m56s	177MB
1h33m17s	85.4MB
1h10m33s	64.5MB
1h02m08s	56.8MB
31m26s	28.7MB

Tabla 3.9 Archivos De Audio Programación

La Tabla 3.10 muestra los tiempos empleados, esta medición se la realiza con el temporizador interno de Matlab al hacer uso de funciones *tic* y *toc*.

Duración de Programación	Tiempo de Carga y Fragmentación (segundos)
3h13m56s	39.139570
1h33m17s	11.171111
1h10m33s	7.145732
1h02m08s	6.363579
31m26s	3.127003

Tabla 3.10 Tiempo de Carga y Fragmentación

Los datos mostrados en la tabla anterior podrían variar dependiendo del uso del procesador en la computadora. La prueba de medición del Tiempo de Carga y Fragmentación, se la realizó con la ejecución única de Matlab en Windows 7.

3.2.3.2. Cotejamiento

Se realiza las pruebas con los siguientes archivos.

	Audio Programación	Duración (minutos)
A	20120208_1947_CH8_CH 8.wav	62
B	20120224_0138_CH13_CH 13.wav	63
C	20120308_2001_CH8_CH 8.wav	60

Tabla 3.11 Archivos Programación elegidos para las pruebas.

Audio Spot	Duración (segundos)	Audio Spot	Duración (segundos)
Asepsia.wav	20	Girasol.wav	17
Atun Isa.wav	14	Hlnds.wav	18
Avon.wav	19	Lotto.wav	30
Axe.wav	19	Marcimex.wav	19
BanPich.wav	43	Mensajea.wav	60
BigCola.wav	9	Movistar.wav	10
Chaide.wav	22	NatGard.wav	32
Cicatricure.wav	20	OpGMO.wav	25
Claro.wav	26	Pañalin.wav	20
Colineal.wav	28	Pepsjumb.wav	29
CoopNacional.wav	16	Pozogem.wav	26

CruzAzul.wav	26	RespMejor.wav	28
Diners.wav	24	RevCiu.wav	5
Experta.wav	33	Scot.wav	20
Favorita.wav	29	Sensodyne.wav	29

Tabla 3.12 Archivos Spot elegidos para pruebas

Se hace un cotejamiento tomando un archivo Audio Programación para ir correlacionando con cada uno de los Audios Spots elegidos para las pruebas. Luego de esto se toma el siguiente Audio Programación para repetir el proceso. Al final del cotejamiento se obtiene los resultados expuestos en las siguientes tablas.

Cotejamiento A		
Programación: 20120208_1947_CH8_CH 8.wav		
Spot	Apariciones	Tiempo de Cotejamiento (s)
Asepsia.wav	0	32.14
Atunlsa.wav	0	32.41
Avon.wav	0	32.15
Axe.wav	0	32.21
BanPich.wav	0	32.01
BigCola.wav	0	50.50
Chaide.wav	0	32.60
Cicatricure.wav	0	32.37
Claro3.wav	0	34.10
Colineal.wav	0	32.34
CoopNac.wav	2	32.62
CruzAzul.wav	1	32.31
Diners.wav	1	32.53
Experta.wav	0	33.17
Favorita.wav	0	32.86
Girasol.wav	1	32.49
Hinds.wav	0	33.11
Lotto.wav	0	32.80
Marcimex.wav	0	32.31
Mensajea	0	32.15
Movistar	0	50.94
NatGard.wav	0	32.28
OpGMO.wav	1	33.02
Pañalin.wav	0	32.09
PepsiJum.wav	0	32.81
PozoGem.wav	0	32.14
RevCiud.wav	0	48.38
RespMej.wav	0	32.06
Scot.wav	1	32.19
Sensodyne.wav	1	32.61

Tabla 3.13 Cotejamiento A

Cotejamiento B		
Programación: 20120224_0138_CH13_CH 13.wav		
Spot	Apariciones	Tiempo de Cotejamiento (s)
Asepsia.wav	0	32.76
AtunIlsa.wav	0	32.87
Avon.wav	0	31.94
Axe.wav	0	32.57
BanPich.wav	0	32.46
BigCola.wav	0	51.16
Chaide.wav	0	33.51
Cicatricure.wav	0	32.79
Claro3.wav	0	32.39
Colineal.wav	0	33.24
CoopNac.wav	0	32.90
CruzAzul.wav	0	32.78
Diners.wav	0	33.46
Experta.wav	0	33.62
Favorita.wav	0	33.62
Girasol.wav	0	33.13
Hinds.wav	0	33.93
Lotto.wav	0	33.11
Marcimex.wav	0	32.72
Mensajea.wav	2	32.96
Movistar	0	51.07
NatGard.wav	3	32.85
OpGMO.wav	0	33.73
Pañalin.wav	0	32.24
PepsiJum.wav	0	33.68
PozoGem.wav	0	32.66
RespiraMejor.wav	0	32.88
RevCiud.wav	0	49.08
Scot.wav	0	32.65
Sensodyne.wav	0	32.81

Tabla 3.14 Cotejamiento B

Cotejamiento C		
Programación: 20120308_2001_CH8_CH 8.wav		
Spot	Apariciones	Tiempo de Cotejamiento (s)
Asepsia.wav	0	36.80
AtunIlsa.wav	1	38.70
Avon.wav	0	37.62
Axe.wav	0	37.20
BanPich.wav	0	37.72
BigCola.wav	0	61.22
Chaide.wav	1	38.39
Cicatricure.wav	0	39.77
Claro3.wav	0	37.41

Colineal.wav	0	37.58
CoopNac.wav	2	38.19
CruzAzul.wav	0	32.26
Diners.wav	0	39.03
Experta.wav	0	38.78
Favorita.wav	0	37.84
Girasol.wav	0	38.27
Hinds.wav	0	37.87
Lotto.wav	0	37.55
Marcimex.wav	0	39.98
Mensajea.wav	0	37.00
Movistar.wav	0	58.37
NatGard.wav	0	37.51
OpGMO.wav	0	37.44
Pañalin.wav	0	38.43
PepsiJum.wav	0	38.12
PozoGem.wav	0	37.26
RespMej.wav	0	37.48
RevCiud.wav	0	55.43
Scot.wav	0	36.95
Sensdyne.wav	0	37.19

Tabla 3.15 Cotejamiento C

En el ANEXO 6.2 se tiene la lista de todos los comerciales que aparecen dentro de los Audios Programaciones de la Tabla 3.11.

3.3. ANALISIS DE RESULTADOS

Para alcanzar los objetivos propuestos en este trabajo, se analiza la detección positiva y los tiempos de respuesta de SpotID.

3.3.1. DETECCION POSITIVA

Luego de las 90 correlaciones, los resultados muestran una eficacia del 100%. Lo que representa que SpotID reconoció todas las apariciones que se deseó encontrar y no encontró algún Falso Positivo (spot erróneo). También se debe considerar que exista la probabilidad que, al terminar el escaneo, no se revelen coincidencias cuando en realidad sí existen, esto se ha declarado como Detección Negativa.

Así, de los 18 Audios Spots que se esperaban hallar, se detectaron todos. Y de los 72 mínimos de errores de detección que pudieran haberse escaneado, no se reportó ni uno.

Cotejamiento	Detección Positiva	Detección Negativa	Falso Positivo
A	9/9	0/9	0/21
B	5/5	0/5	0/25
C	4/4	0/4	0/26
	18/18	0/4	0/72
Porcentaje Total Hallado	100%	0%	0%

Tabla 3.16 Resultado de Cotejamiento

3.3.2. TIEMPOS DE RESPUESTA

Los tiempos de cada cotejamiento se tabulan para luego promediarlos, con el fin de obtener una tendencia del tiempo que emplea SpotID en analizar una propaganda.

Audio Spot	Tiempo de Cotejamiento A (segundos)	Tiempo de Cotejamiento B (segundos)	Tiempo de Cotejamiento C (segundos)
Asepsia.wav	32.14	32.76	30.80
AtunIsa.wav	32.41	32.87	32.70
Avon.wav	32.15	31.94	31.62
Axe.wav	32.21	32.57	31.20
BanPich.wav	32.01	32.46	31.72
BigCola.wav	50.5	51.16	55.22
Cicatricure.wav	32.37	32.79	33.77
Claro.wav	34.1	32.39	31.41
Colineal.wav	32.34	33.24	31.58
CoopNac.wav	32.62	32.9	32.19
CruzAzul.wav	32.31	32.78	26.26
Chaide.wav	32.6	33.51	32.39
Diners.wav	32.53	33.46	33.03
Experta.wav	33.17	33.62	32.78
Favorita.wav	32.86	33.62	31.84
Girasol.wav	32.49	33.13	32.27
HInds.wav	33.11	33.93	31.87
Lotto.wav	32.8	33.11	31.55
Marcimex.wav	32.31	32.72	33.98

Mensajea.wav	32.15	32.96	31.00
Movistar.wav	50.94	51.07	52.37
NatGard.wav	32.28	32.85	31.51
OpGMO.wav	33.02	33.73	31.44
Pañalin.wav	32.09	32.24	32.43
Pepsjumb.wav	32.81	33.68	32.12
Pozogem.wav	32.14	32.66	31.26
RespMej.wav	32.38	32.88	31.48
RevCiu.wav	48.81	49.08	49.43
Scot.wav	32.19	32.65	30.95
Sensodyne.wav	32.61	32.81	31.19

Tabla 3.17 Tiempos de Respuestas por Cotejamiento

En la Figura 3-5 muestra los tiempos promediados de los cotejamientos A, B y C. Así, de manera gráfica puede deducirse que, el tiempo empleado SpotID en buscar un anuncio publicitario de mayor a 20 segundos de duración en una programación de 60 minutos, es de 32 segundos aproximadamente.

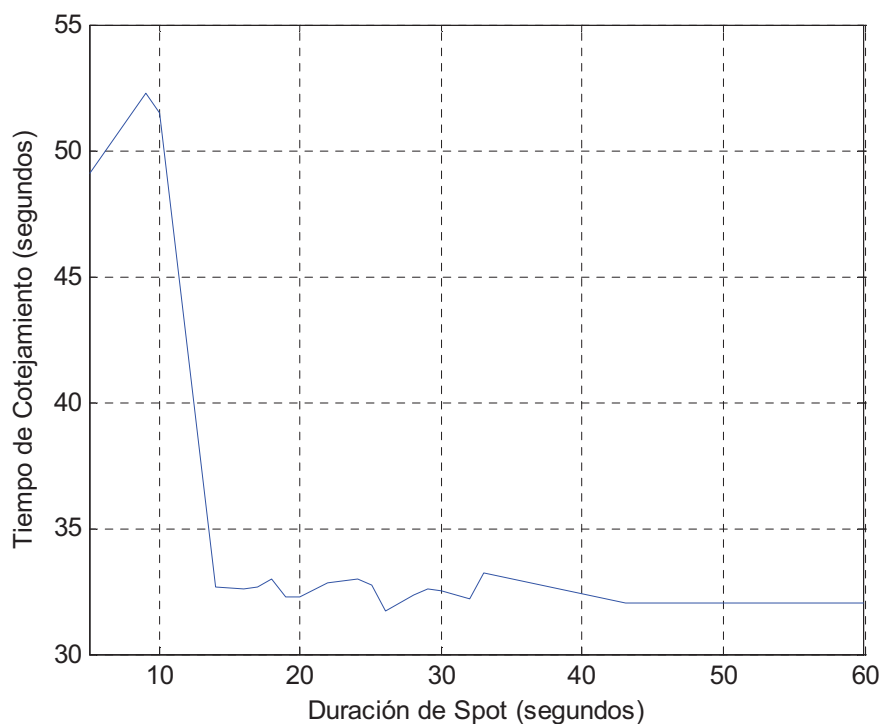


Figura 3-5 Tendencia de Cotejamiento

También en la Figura 3-5 se puede notar que mientras el archivo de spot sea menor a los 15 segundos, SpotID tarda más en procesarlos. Esto se debe exclusivamente a la manera de trabajar la función *xcorr* de Matlab, sin que esto sea causa de falsos positivos.

Sumando el tiempo que conlleva cada proceso del sistema, se puede concluir que: Desde el momento de capturar una hora de programa hasta obtener una respuesta, se emplean alrededor de 65 minutos, según la Tabla 3.18.

Tiempo empleado en Hallar un Spot (minutos)		
Actividad	Manualmente	SpotID
Captura	60	60
Extracción Audios	0	2.5
Búsqueda y Presentación de Resultados	30	0.7
Total	90	63.2

Tabla 3.18 Comparación de Tiempos de Búsqueda

3.4. ANALISIS DE COSTOS DEL PROTOTIPO

Los costos de este sistema están basados en los elementos empleados por el autor al momento del desarrollo, pero se agrega hardware recomendado para la mejor estructura y funcionalidad del sistema. Los costos fueron consultados en internet.

Debido al sigilo con que han mantenido varias empresas propietarias de software que realizan el mismo trabajo, este análisis se lo realiza sin comparación frente a otro sistema. Se agrega los costos de los componentes que conforman el hardware y software inmerso del sistema recomendado para un completo funcionamiento. A esto se agrega el costo de diseño e implementación invertidos en horas por el autor de este trabajo.

3.4.1. COSTOS REFERENCIALES DEL PROYECTO

El cálculo proyectado para realizar este trabajo ascendería al valor de 6722 dólares americanos con 87 centavos de dólar. Desglosado en la Tabla 3.19.

Cabe recalcar que las licencias de los programas computacionales: Windows 7, Office 2007 y Matlab, son autorizados para uso exclusivo con fines investigativo personal, pues las licencias de estos programas para fines comerciales son mucho mayores a los precios mostrados aquí.

El costo de diseño e implementación está basado aproximadamente al trabajo diario de 8 horas, 5 días a la semana durante 3 meses, asignándole US\$10 la hora de trabajo.

Item	Precio
Licencia Windows 7 Ultimate 64 Bits	299.99
Office 2007 Pyme Small Bussines	140.00
Licencia Matlab Student Version	99.00
Zogis 400U Real Angel USB	43.00
Laptop HP Pavillion dv7-3065dx	700.00
Regulador de Voltaje con UPS	48.99
Disco duro externo 1.5 TB Conex Electric	269.99
Antena de Aire VHF/UHF	32.00
Impresora multifuncional. Deskject 350	89.90
Costo de Diseño e Implementación	5000.00
Total	6722.87

Tabla 3.19 Costo del Proyecto

4. CAPÍTULO CUATRO. CONCLUSIONES Y RECOMENDACIONES

En este apartado se expone las conclusiones luego de realizar el diseño, pruebas y resultados del sistema SpotID. Se agregan recomendaciones para encaminar las ideas que futuros interesados puedan tener al respecto de este trabajo para modificarlo en busca de su mejora, suponiendo que el lector de este capítulo tenga conocimientos en informática y tecnologías de la información.

4.1. CONCLUSIONES

- Basado en los resultados de las pruebas, se puede concluir que el objetivo de reconocer publicidad se cumple. Se destaca el grado de asertividad de un 100%, reforzando el hecho con las decenas de pruebas realizadas durante el desarrollo y que no han sido transcritas en este trabajo escrito.
- Además, el prototipo cumple con el segundo objetivo propuesto, que es la optimización del tiempo; pues, realiza el mismo trabajo de un ser humano pero en menor tiempo.
- Aunque, durante las pruebas de SpotID, se manifiesta que el cerebro humano logra rebasar a la máquina, puesto que en menos de diez segundos, el hombre puede detectar y reconocer varias imágenes de comerciales al mismo momento en que las está viendo, aunque esta sea, la primera vez que lo vea. El desafío para el cerebro empieza cuando tiene que revisar videos de más de dos horas de duración.
- La calidad de grabación del audio tiene un efecto mínimo. Según pruebas durante el diseño, se obtienen los mismos resultados si se procesan archivos con mayor velocidad de bits a 8kbps (audios con mayor calidad) lo que también aumenta el tiempo de procesamiento y el mayor consumo de memoria. Cabe recalcar que SpotID sólo trabaja con registros de 8kbps monofónicos.
- El mismo efecto se tendría si calidad de la grabación se la registra en modo estéreo; sólo aumentaría el tiempo de procesamiento y mayor consumo de memoria de almacenamiento.

- El resultado de la correlación cruzada son valores negativos y positivos, en este proyecto la correlación cruzada es presentada como valor absoluto, sin ver afectación alguna en el cotejamiento y los resultados.
- La correlación cruzada realiza desplazamiento del tiempo hacia adelante y atrás del archivo AS, solo es necesario para los tiempos positivos debido a que la muestra del spot únicamente se lo hace en función de la programación en adelante.
- El tiempo de captura de los AP debe ser menor a 12 horas para facilitar el manejo y procesamiento de estos archivos. Aunque si el poder de procesamiento del computador lo permite, Spot ID está en la capacidad de manejar el tamaño máximo de un archivo WAV que es alrededor de 4.3GB.
- Al tener una confiabilidad mejor a la esperada y una optimización del tiempo competitiva frente al pensamiento humano, se puede concluir que tiene viabilidad para su mejoramiento y comercialización.

4.2. RECOMENDACIONES

- Para obtener una base de datos de spots, o sea los AS, con la mayor cantidad de muestras y de buena calidad, se puede optar por solicitar los archivos de audio original de la empresa anunciante, o en su defecto, editar los audios de radiodifusoras de excelente calidad.
- Para trabajos futuros se debería estudiar una manera de normalizar los resultados de la correlación. Ya que en este trabajo se ve limitado por el tipo de estructura del mismo. Para ser entendible un margen entre 0 y menos uno.
- Con el fin de optimizar espacio en memoria y el tiempo de procesamiento, es válido buscar un formato de audio más compacto o comprimible. Uno de estos formatos es el MP3. Cabe recordar que MATLAB trabaja únicamente con audios en formatos WAV y AAC. Por tal razón se podría buscar o crear subrutinas que permitan trabajar con archivos de formato MP3. Al trabajar con estos archivos se debería revisar si el tiempo de procesamiento vale la pena el esfuerzo de trabajo con un MP3.

- Un proceso posterior al haber encontrado spots, es poder discernir si el spot hallado, cumple con la duración y calidad pactada entre la empresa anunciante (dueña del comercial) y la estación televisiva.
- Se debería integrar al sistema SpotID la tarea de extracción de audio. Así, el tiempo de procesamiento podría disminuir prescindiendo de los programas Audacity y DVDVideoSoft Free Studio.
- Debido a las posibles capturas diarias de AP y AS, se recomienda organizar los archivos, de tal manera que se pueda encontrar los archivos por canal, fecha, horario o programa difundido.
- Tomando en cuenta que los nombres de los archivos de audio tienen una nomenclatura similar a un código numérico (20120308_2001_CH8_CH8.wav), se recomienda al momento de almacenarlos en las respectivas carpetas, renombrar con nombres comprensibles por el operador de SpotID; por ejemplo Ecuavisa026.
- Se aconseja realizar la captura de tres archivos diarios, en los horarios de 00h00 a 08h00, de 08h00 a 16h00 y de 16h00 a 24. Aunque queda discreción del poder de procesamiento del computador.
- Una notable mejora a este proyecto, es la posibilidad de escanear spots en el mismo momento que se está capturando el video de programación o con un pequeño “delay” (retraso de tiempo para empezar a realizar una tarea) luego de la captura.
- SpotID muestra datos de los archivos que se están cotejando, pero se podría ampliar la información con apoyo visual, así el operador podría verificar los archivos que se están procesando mediante videos de pequeña duración o bastaría con imágenes, de igual manera pueden ser tratados los resultados.
- Para optimizar el tiempo de procesamiento, dentro de SpotID, se puede mejorar la búsqueda al realizar varios AS a la vez, pues la búsqueda de spots se la realiza uno por uno.
- Otra opción para optimizar el tiempo de búsqueda, se debería realizar el cotejamiento en paralelo ejecutando multis Sesiones de SpotID. Se puede construir programas ejecutables de SpotID con el compilador de Matlab

(herramienta computacional que traduce el lenguaje de programación en un programa ejecutable sin necesidad de tener instalado Matlab).

- Una rutina también aceptable en SpotID es escanear y encontrar en los AP el momento que empiezan los mensajes comerciales. Así, SpotID escaneará exclusivamente las franjas o momentos de los comerciales del archivo AP. Se pueden basar la detección en el tipo de sonidos, que por lo general, en noticieros, novelas o películas los registros sonoros son diferentes a las características sonoras de los spots.
- Aunque aquí no tiene mayor incidencia los efectos de corriente DC, sería conveniente estudiar a detalle el efecto del desplazamiento DC. Así mismo el desplazamiento de la corriente AC de la fuente, en este caso de 60hz y Ruidos causados por motores y de más artefactos eléctricos.
- Se debe tener en cuenta la protección de un pararrayos, debido a que la computadora está directamente acoplada, mediante el capturador de video, a una antena de aire propensa a una descarga electrostática de un rayo.
- La tarjeta capturadora utilizada en este trabajo puede ser reemplazada con alguna otra con mayores prestaciones, como por ejemplo la grabación de más de un canal a la vez. Habrá que tomar en cuenta que el ZOGIS Real Angel 400U Dongle USB 2.0 PRO provee un nombre único a los archivos grabados y ese formato es procesado dentro de la codificación de SpotID. Por lo que deberá tomarse en cuenta que con otro tipo de nomenclatura la información de AS y AP será errónea.
- Con el inicio de la transmisión de televisión de contenido digital, tal vez, el trabajo de SpotID ya no sería cotejar audios, sino que, se vería encausado a la búsqueda, directamente dentro de la señal modulada, algún indicio que informe la presencia de un spot publicitario.

5. REFERENCIAS

¹ http://es.wikipedia.org/wiki/Mercado_objetivo

² John G. Proarkis, D. G. (1998). Tratamiento Digital de Señales. Madrid: Prentice Hall.

³ <http://www.promonegocios.net/mercadotecnia/publicidad-definicion-concepto.html>

⁴ orbitmedia.com.mx/Mon_Publicitario.htm

⁵ http://es.wikipedia.org/wiki/Claude_Elwood_Shannon

⁶ http://es.wikipedia.org/wiki/Waveform_Audio_Format

⁷ http://es.wikipedia.org/wiki/Capturadora_de_televisi%C3%B3n

⁸ <http://zogis.com/index.php?cont=products&id=243>

⁹ <http://www.dvdvideosoft.com/es/products/dvd/Free-Video-to-MP3-Converter.htm>

¹⁰ <http://www.mathworks.es/products/matlab/>

¹¹ Bertran, Eduard. (2003). Señales y Sistemas de tiempo discreto. Ediciones UPC

¹² <http://www.mathworks.es/help/toolbox/signal/ref/xcorr.html>

¹³ Especificación de Requisitos según el estándar de IEEE 830

¹⁴ Recomendación UIT-T J.62

6. ANEXOS

6.1. MANUAL DEL USUARIO



MANUAL DEL USUARIO



SpotID, la herramienta de apoyo para el monitoreo de publicidad en radio o televisión. Puede escanear 12 horas de programación en busca de algún spot publicitario y arrojar resultados en menos de 6 minutos.

Contenido

Requerimientos Mínimos de Hardware y Sistema Operativo.	69
Introducción.....	69
Carga de archivos.....	70
• Selección de Audio Programación.	71
• Selección de Audio Spot.....	72
Cotejamiento	73
Resultados.....	74
• Imprimir Reporte	75

Requerimientos Mínimos de Hardware y Sistema Operativo.

Para el uso de SpotID se debe considerar como mínimo los siguientes requerimientos:

- Intel Pentium-IV 2.0GHz, procesador AMD equivalente o superior.
- 2GB de memoria RAM.
- Puerto USB 2.0.
- Tarjeta gráfica con 64 MB de memoria (compatible con DirectX 9.0c).
- Tarjeta de sonido compatible con AC97.
- 2 GB de espacio libre en el disco duro.
- Unidad de CD-ROM (para la instalación de software adicional).
- Sintonizador y Capturador ZOGIS Real Angel 400U Dongle USB 2.0 PRO.
- Antena de aire Señal de Operador de Cable.
- Microsoft Windows® XP SP2/XP MCE 2005/Windows Vista.
- Microsoft Windows Excel 2007.
- Software de extracción de audio a formato WAV con 8kbps monocal.
- Matlab® R2007a.

Introducción

Este manual le permitirá aprender a utilizar las funcionalidades básicas de SpotID. SpotID funcionará exclusivamente con archivos que guardan un formato pre-establecido e están indicados en este manual.

El funcionamiento consta de tres pasos:

- Carga de archivos
 - Selección de Audio Programación
 - Selección de Audio Spot
- Cotejamiento
- Resultados
 - Impresión de resultados

Carga de archivos

Antes asegurarse que los archivos de audio a cargarse estén grabados en formato WAV y cumplir con las siguientes características:

- Audio Spot (AS) menor a 1 minuto de duración,
- Audio Programación (AP) menor a 12 horas¹
- Monocanal; y,
- Velocidad de bits 64kbps.

Los archivos AP y AS pueden estar localizados en cualquier ubicación del computador. Se recomienda mantener almacenado los archivos en una sola dirección aunque en carpetas separadas indicando los archivos AP y AS.

Tomando en cuenta los la configuración de AS y AP, se procederá a cargar en SpotID.

El software tiene una ventana principal (Figura 6.1), en la cual es visible los botones activos para empezar el uso.

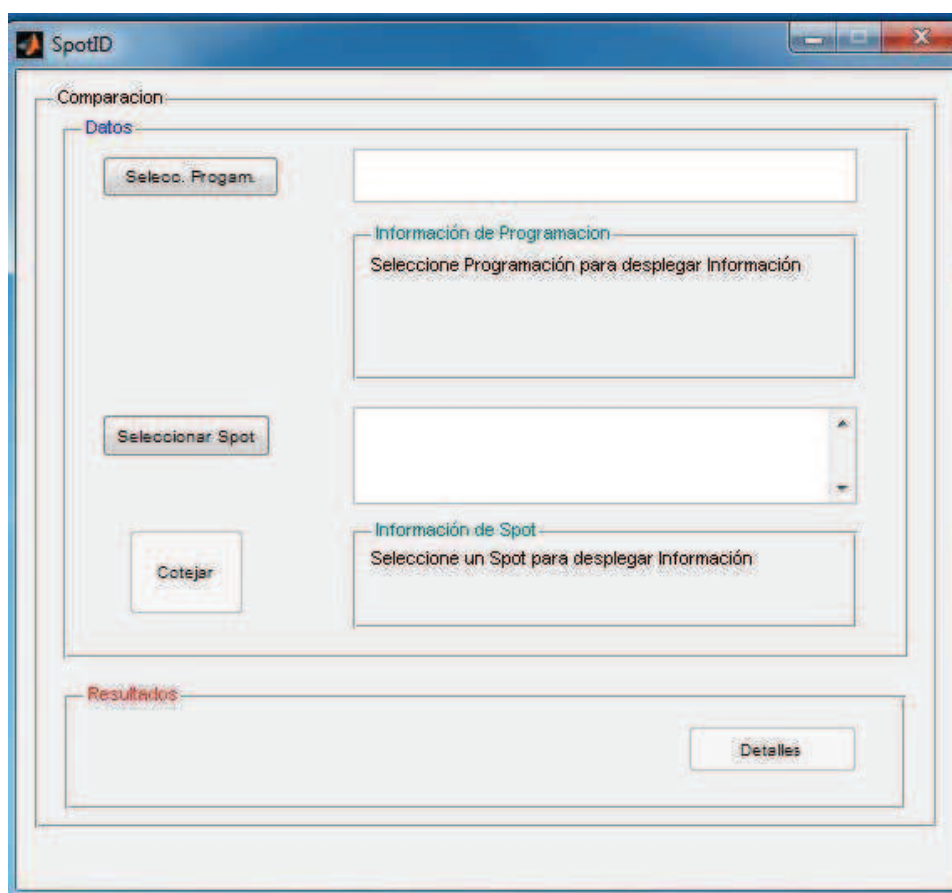
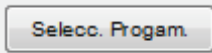


Figura 6.1 Ventana Principal de SpotID

¹ Según las pruebas realizadas, procesa 12 horas en menos de 6 minutos, aunque está en capacidad de procesar audios de programación mas allá de las 12 horas, pero ralentizaría el sistema.

- **Selección de Audio Programación.**

Al dar clic sobre el botón , se abrirá una ventana como la Figura 6.2 para escoger un AP.

Se escoge un archivo que desee escanear, y luego se da clic en el botón Abrir.

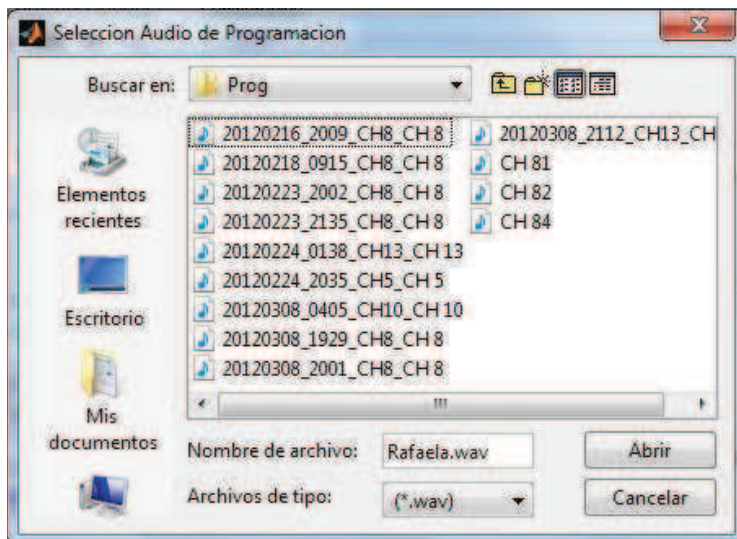


Figura 6.2 Selección de AP

Inmediatamente empieza a cargarse el Archivo Programación, mientras dure este proceso se verá una ventana de avance de carga con la leyenda "Fragmentando Programación". Figura 6.3

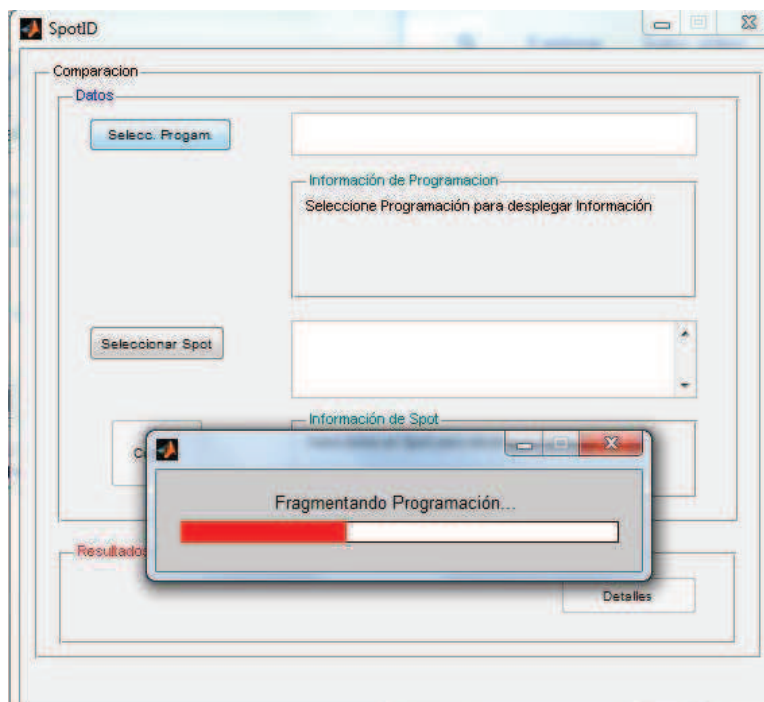


Figura 6.3 Avance de carga y fragmentación

Si el archivo se cargó correctamente se podrá ver información del AP como lo muestra la Figura 6.4.

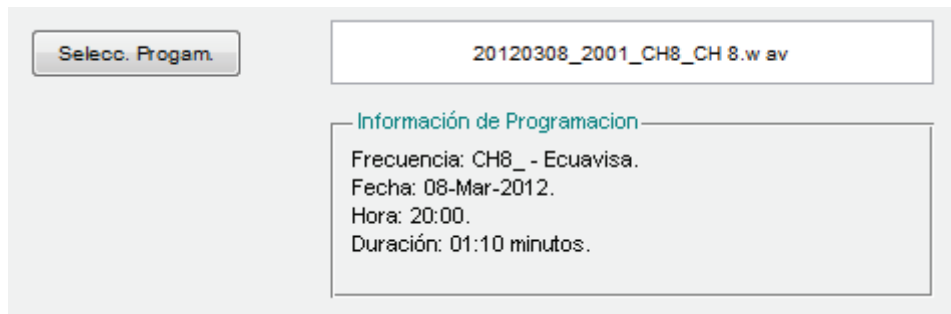


Figura 6.4 Información de Archivo Programación

Luego se procede a cargar el AS.

- **Selección de Audio Spot.**

Para cargar el AS se presiona el botón

Luego se abre la ventana de “Selección de Audio Spot” Figura 6.5, se escoge algún archivo AS y se da clic en Abrir.

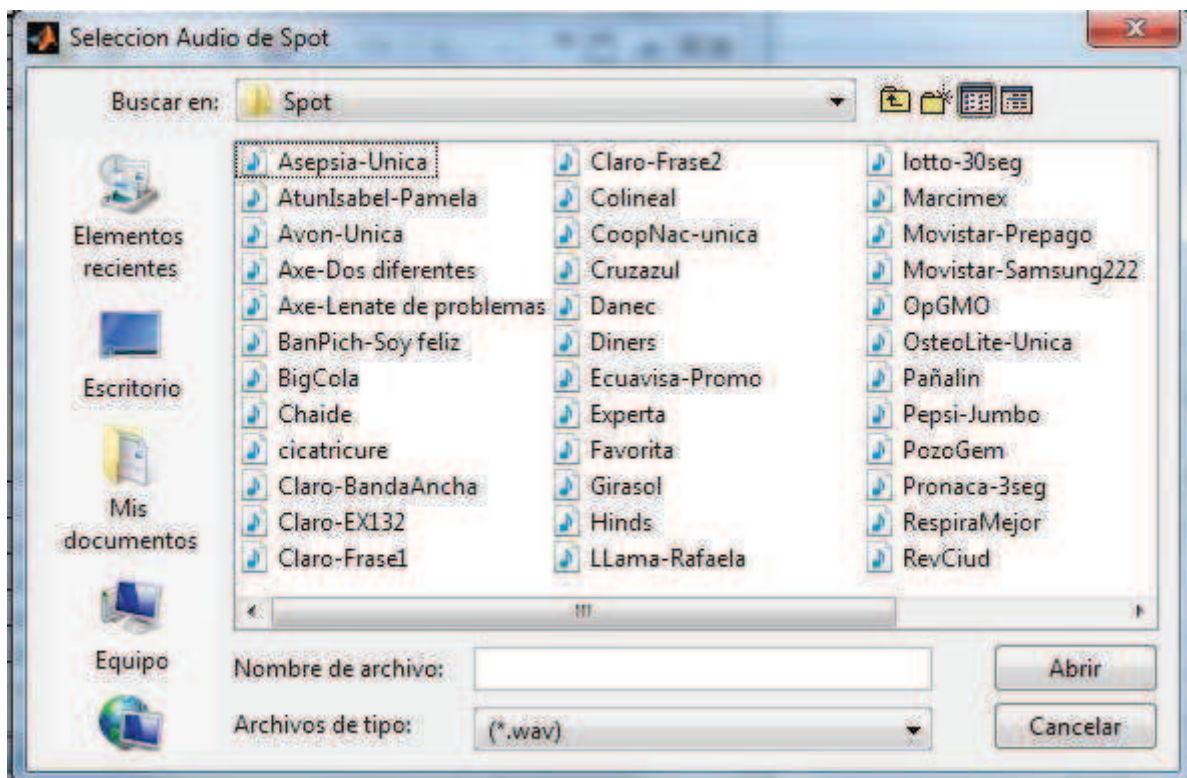


Figura 6.5 Selección AS

Para asegurarse de que el AS está cargado el Audio Spot se tiene que ver información del spot, como lo muestra la Figura 6.6

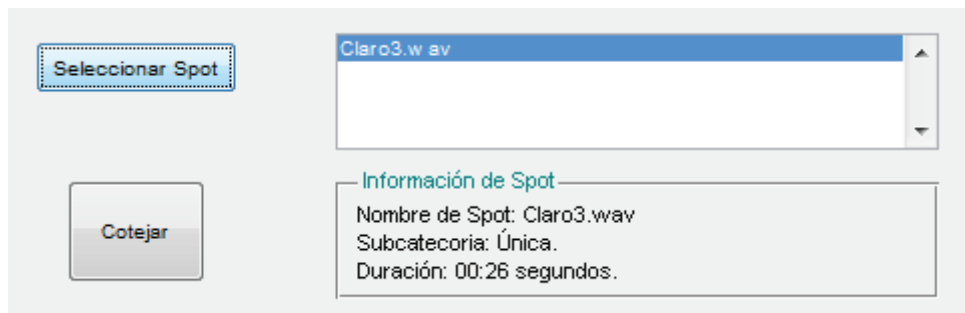
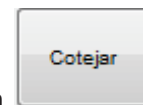


Figura 6.6 Información de Archivo Spot

Al darle un click sobre el nombre del spot abierto y cargado, se reproducirá 5 segundos iniciales del audio spot.

Cotejamiento



Ya cargados al programa los archivos AP y AS, se puede dar clic en el botón

Aparecerá en pantalla una barra de avance del proceso, Figura 6.7 .Una vez terminado se podrá optar por ver el informe.

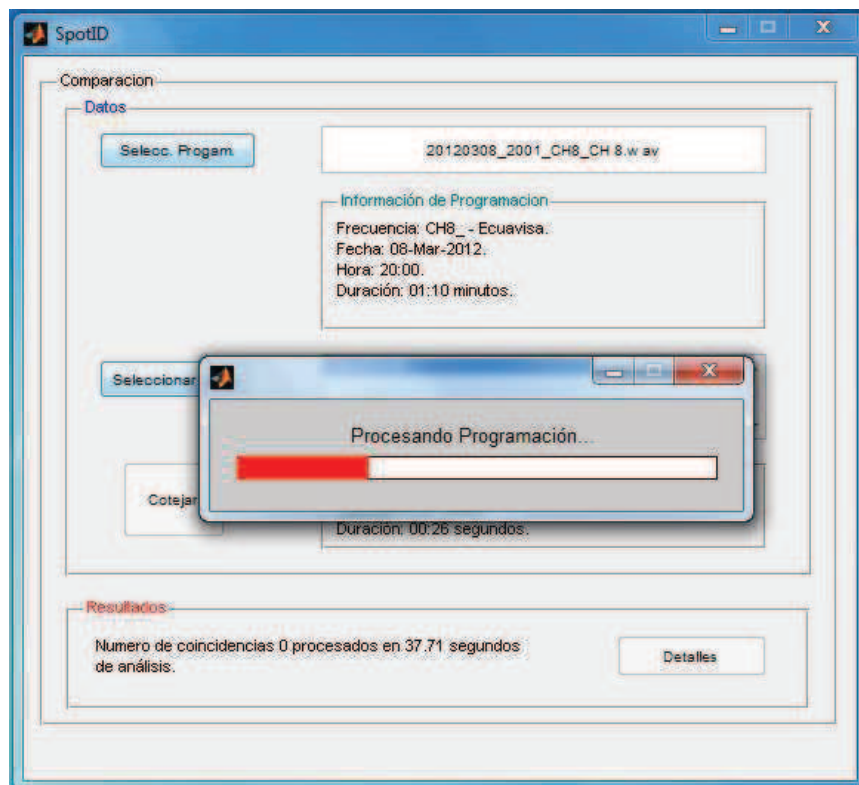


Figura 6.7 Momento de Procesamiento de Audios Programación y Spot

Resultados

Los resultados del cotejamiento pueden verse preliminarmente al pie de la ventana.

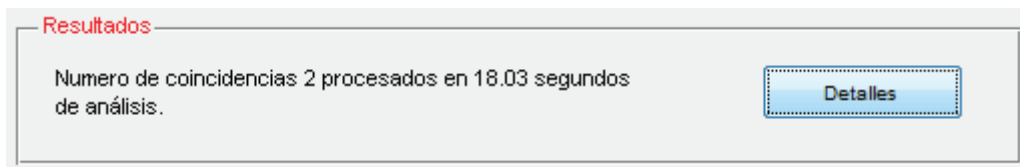
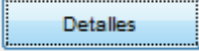


Figura 6.8 Pie de Ventana en la que se muestra brevemente el resultado

Al dar clic en el botón , generará la aparición de una pequeña ventana "Resultados" mostrando los tiempos en que se hallaron semejanza, si existieran, del AS en el Audio Programación, Figura 6.9.

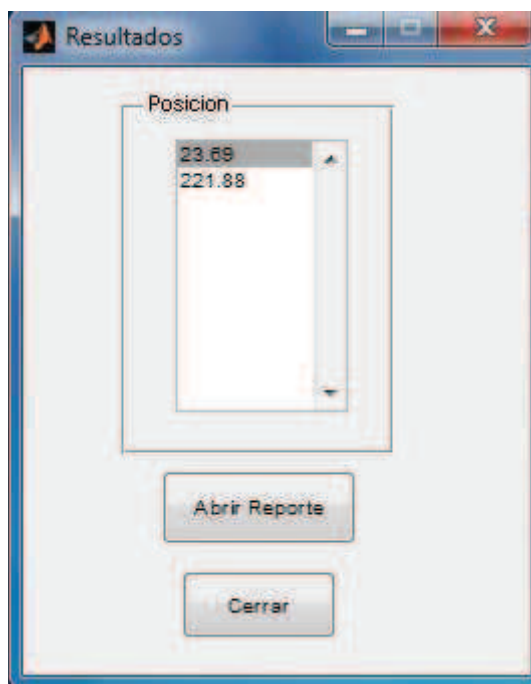


Figura 6.9 Ventana Resultados

En la ventana de "Resultados", se puede acceder al reporte en Excel para poder visualizarlo y siendo opcional de imprimirlo. Figura 6.10.

6.2. DETALLES DE PROGRAMACIÓN DE AP

	Audio Programación	Duración (minutos)
A	20120208_1947_CH8_CH 8.wav	62
B	20120224_0138_CH13_CH 13.wav	24
C	20120308_2001_CH8_CH 8.wav	60

A: 20120208_1947_CH8_CH 8.wav		
Hora	Programa	Apariciones
0:00	Novela	1
0:28	Ecuador Tiene talento	1
1:11	Ranchero	1
1:37	Sensodyne	1
2:07	LadySpeedStick	1
2:28	Mr Cook	1
2:43	CNT	1
3:14	Royal	1
3:24	Decisiones	1
9:03	Televistazo	1
19:03	Contacto Directo	1
20:31	Computer Power	1
21:03	LanPass	1
21:19	Esika	1
21:29	Atun Real Baile	1
22:10	Super Cone	1
22:26	Televistazo	1
30:31	Cuchicheos	1
31:07	Asi Somos	1
31:32	CoopNacional	1/2
31:50	Rey Queso	1
32:11	Lava Todo	1
32:32	Girasol	1
32:49	Diners	1
33:14	Televistazo	1
40:19	Telenovela	1
40:48	CruzAzul	1
41:13	Universo Postres	1
41:34	Scot	1
41:55	GMO	1
42:20	CoopNacional	2/2
42:38	Televistazo	1
53:28	Novela	1

B: 20120224_0138_CH13_CH 13.wav		
Hora	Programa	Apariciones
00:00	Noticiero	1
00:19	Danilo Rosero	1
1:00	NatureGarden	1
1:22	BioCapill	1
2:31	Mensajea	1
3:35	Osteolite	½
4:10	Noticiero	1
11:34	Bailalo	1
11:54	YingYang Kramery	1
12:18	Inau Kramery	1
12:35	Talisman Kramery	1
13:08	Tarot Kramery	1
13:24	Osteolite	2/2
13:57	Eucamiel	1
14:34	Divinas	1
15:14	Diosas	1
15:37	Noticiero	1
20:27	Bailando	1
20:58	Onella	1
21:59	Eucamiel	1
22:38	Tweeze	1
23:35	GardenLact	1
24:09	Falla de sintonía	1

C: 20120308_2001_CH8_CH 8.wav		
Hora	Programa	Apariciones
00:00	Televistazo	1
8:33	Contacto Directo	1
9:00	Mr Chancho	1
9:30	Universidad Ecotec	1
9:46	EspecialK	1
10:07	Chili's	1
10:25	Stevia Life	1
10:55	Banco Pacifico	1
11:25	Ecomundo	1
11:56	Televistazo	1
20:07	Cuchicheos	1
20:44	CoopNacional	1/2
21:02	Optica Andes	1
21:17	Atunlsabel	1
21:32	DirectTV	1
22:05	Clorox	1
22:19	Almohada Chaide	1
22:40	Girasol	1
23:11	Kotex	1
23:47	Televistazo	1
30:58	La casa de Alado	1
31:18	Asi Somos	1
31:32	Progress Gold	1
32:02	Chevrolet	1
32:31	CoopNacional	2/2
32:49	Universidad Casa Grande	1
32:58	Western Union	1
33:27	Tarifa Multi Claro	1
33:43	UV GMO	1
34:09	Discover	1
34:42	Televistazo	1
44:56	Cuchicheos	1
58:27	Revolución Ciudadana	1

6.3. CODIGO FUENTE

```

%Inicialización del archivo SpotID.m

function varargout = untitled(varargin)

%Last Modified by GUIDE v2.5 07-Mar-2012 17:50:11

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @untitled_OpeningFcn, ...
                  'gui_OutputFcn',  @untitled_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before untitled is made visible.
function untitled_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to untitled (see VARARGIN)

% Choose default command line output for untitled
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes untitled wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = untitled_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure

%Declaración de variables que se usaran en todo el programa
global k tit dat %S

```

```

varargout{1} = handles.output;
k=1;          %Contador General
tit=[];      %Cadena Alfanumerica de información del AP
dat=[];      %Cadena Alfanumerica de información del AS
%Resetea datos de Archivo Excel
dat0=zeros(10);
    xlswrite('datos', dat0,'Prog','A2');
dat0=zeros(16,5);
    xlswrite('datos', dat0,'Spot','A2');

%--- Executes during object creation, after setting all properties.
function listbox1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%         See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in AbrPro.
function AbrPro_Callback(hObject, eventdata, handles)
% hObject    handle to AbrPro (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global Fs PP F P ProgName tit a

%Seleccion del archivo del AP
[ProgName Path]=uigetfile({'*.wav'}, 'Seleccion Audio de Programacion');
if isequal(ProgName,0)
return
else
    [P Fs]=wavread(strcat(Path,ProgName)); %Lectura del AP
end
%extrae Canal
ch=ProgName(15:18);

%Establece Nombre Canal
if ch=='CH2_', Estacion='GamaTV'; end
if ch=='CH4_', Estacion='Teleamazonas'; end
if ch=='CH5_', Estacion='RTS'; end
if ch=='CH7_', Estacion='ECTV'; end
if ch=='CH8_', Estacion='Ecuavisa'; end
if ch=='CH10', Estacion='TC'; end
if ch=='CH13', Estacion='Canal Uno'; end

%extrae fecha archivo
a=[str2num(ProgName(1:4)) str2num(ProgName(5:6)) str2num(ProgName(7:8))...
    str2num(ProgName(10:11)) str2num(ProgName(11:12)) 0];
aaa=str2num(ProgName(10:11))+(str2num(ProgName(11:12))/60);
fecha=datestr(a,1);

```



```

%hora de inicio
hini=datestr(a,15);

%duracion
d=(length(P)/69120000);% para formato minutos/segundos
duracion=datestr(d,15);

%hora fin
e=datenum(a)+d;
hfin=datestr(e,13);

%establece Horario A,AA,AAA
if (aaa>19)&&(aaa<21.5)
    Sintonia='AAA';
elseif (aaa>15)&&(aaa<19)
    Sintonia='AA';
else
    Sintonia='A';
end

%PP es la fragmentación del AP en matriz de F columnas de 2,18seg de audio,
%F Número de Fragmentos
[PP,F]=fragmen(P);
frag=num2str(F);

%Fecha de escaneo
fechascan=datestr(now,0);

%muestra Informacion del Programacion
set(handles.ProgTxt,'String',ProgName);
info=sprintf('Frecuencia: %s - %s.\nFecha: %s. \nHora: \t%s....\nDuración: %s minutos.',ch,Estacion,fecha,hini,duracion);
set(handles.InfoProg,'String',info);
%Ingresa datos para el reporte
tit=[tit;{Estacion fecha hini hfin duracion ch Sintonia fechascan frag}];
xlswrite('datos', tit,'Prog','A2');

function ProgTxt_Callback(hObject, eventdata, handles)
% hObject    handle to ProgTxt (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of ProgTxt as text
%        str2double(get(hObject,'String')) returns contents of ProgTxt as a
double

% --- Executes during object creation, after setting all properties.
function ProgTxt_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ProgTxt (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

```

```

%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in AbrSpt.
function AbrSpt_Callback(hObject, eventdata, handles)
% hObject    handle to AbrSpt (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global Fs S k SpotNameO subcat
[SpotNameO Path]=uigetfile({'*.wav'}, 'Seleccion Audio de Spot');
if isequal(SpotNameO,0)
return
else
    [S Fs]=wavread(strcat(Path,SpotNameO));
    % [SS Fs]=wavread(strcat(Path,SpotName));
end
%genera matriz de spots de 5 seg

%S(:,k)=SS(end-79999:end);
%length(S)
k=k+1;
SpotList=get(handles.SptLst,'String');
SpotName=[{SpotNameO};SpotList];

d=(length(S)/11520000);
duracion=datestr(d,15);

set(handles.SptLst,'String',SpotName);
set(handles.Cotejar,'Enable','on');
posprog=sprintf('Cotejar para presentar resultados');
set(handles.ResTxt,'String',posprog);
info=sprintf('Nombre de Spot: %s \nSubcatecoria: %s.\nDuración: %s
segundos.',SpotNameO,subcat,duracion);
set(handles.InfoSpot,'String',info);

% --- Executes during object creation, after setting all properties.
function SptLst_CreateFcn(hObject, eventdata, handles)
% hObject    handle to SptLst (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in Cotejar.
function Cotejar_Callback(hObject, eventdata, handles)
% hObject    handle to Cotejar (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global PP S pos F SpotNameO subcat dat a%ValMaxCorr
%clear pos
set(handles.Cotejar,'Enable','off');
set(handles.Detalles,'Enable','off');

delete(resultados)
tic
%Carga de archivos

h = waitbar(0,'Procesando Programación...');
%correlaciona prog vs spot,
ValCorr=[0; 0; 0];

% computation here %

for i=1:F
    Val=correla(PP(:,i),S(:,1));
    ValCorr=[ValCorr ;Val];
    waitbar(i/F)
end
close(h)

y=max(ValCorr);
ValCorr=abs(ValCorr/y);
i=0;
if y > 1500 %1500 valor minimo de correlacion obtenido en pruebas
    ValCorr=ValCorr>0.5;
    pos=find(ValCorr); %pos devuelve ubicacion de las coincidencias
    l=length(pos);
    if l>1
        posrep=floor(pos/480000);
        while i~=l-1
            l=length(pos);
            posrep=floor(pos/480000);
            for i=1:l-1 %elimina las correlaciones efectivas cercanas
                if posrep(i+1)==posrep(i)
                    pos(i+1)=[];
                    break
                end
            end
        end
    end
else
    ValCorr=[0; 0; 0];
    pos=find(ValCorr);
end

toc
t=toc;
if max(pos)>1
    set(handles.Detalles,'Enable','on');
end
posprog=sprintf('Numero de coincidencias %d procesados en %.2f segundos...
de análisis.',length(pos),t);%numero de veces
xlswrite('datos', t,'Prog','J2');
set(handles.ResTxt,'String',posprog);
set(handles.Cotejar,'Enable','on');
beep

```

```

aa=datenum(a);
dura=length(S(:,1))/691200000;%realizado para 8000fps
posx=aa+(pos/691200000);
if isempty(pos)
    dat=[dat;{SpotNameO},dura,length(pos),{'No hay pautas'}];
    xlswrite('datos', dat,'Spot','A2');
else
    for i=1:length(pos)
        dat=[dat;{SpotNameO},dura,length(pos),posx(i)];
        xlswrite('datos', dat,'Spot','A2');
    end
end
%clear

%save ('PPP.mat', 'P')
% --- Executes on selection change in SptLst.
function SptLst_Callback(hObject, eventdata, handles)
% hObject    handle to SptLst (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns SptLst contents as cell
array
%         contents{get(hObject,'Value')} returns selected item from SptLst
global S Fs
inf=get(hObject,'Value');
gos=get(hObject,'String');

switch inf
case inf
    %clear playsnd
    wavplay(S(1:24000),inf),Fs)
end
guidata(hObject,handles);

% --- Executes on button press in Detalles.
function Detalles_Callback(hObject, eventdata, handles)
% hObject    handle to Detalles (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
resultados

function y = correla(A,B)
%Correlación
%Al igual que la convolución, la correlación es una
%operación básica del procesamiento de imágenes digitales.
%La correlación es la operación básica en los
%procesos de búsqueda de patrones por emparejamiento.
y=xcorr(A,B);
%se ajusta a la longitud del vector programación
%xcorr devuelve variable de longitud duplicada
%al vector de mayor longitud
y(1:length(A))=[];

%yy=y.^2;

```

```

%yy=sum(yy);
%yy=sqrt(yy);
%y=y/yy;
%stem(y);
%ValMaxCorr guarda un vector logico en donde
%se encuentre la maxima correlacion
%ValMaxCorr=y>1;

%fragmentacion de un archivo de audio de mayor a 1048576 muestras
function [z,F]= fragmen (A)
h = waitbar(0,'Fragmentando Programación...');
%Toma la longitud total del archivo de audio
L=length(A);

%LS longitud de archivos fragmentados
LS=1048576;

%F numeros de fragmentos de longitud LS
F=floor(L/LS);

%M sobra o módulo del audio Programacion
M=mod(L,LS);

%z Matriz de Audio fragmentado en LSxL+1
z(1:LS,1)=A(1:LS); %Fragmento 1º
for i=1:F-1
    z(:,i+1)=A((LS*i)+1:LS*(i+1)); %Fragmentos 2º,3º,...,iº
    waitbar(i/(F-1))
end
close(h)
z(1:M,F+1)=A(L-M+1:end); %Fragmento último

%numero de columnas o fragmentos
F=F+1;

function varargout = resultados(varargin)
% Last Modified by GUIDE v2.5 01-Mar-2012 02:18:38

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @resultados_OpeningFcn, ...
    'gui_OutputFcn', @resultados_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before resultados is made visible.

```

```

function resultados_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to resultados (see VARARGIN)

% Choose default command line output for resultados
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes resultados wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = resultados_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global pos
% Get default command line output from handles structure
varargout{1} = handles.output;
postxt=num2str(pos/48000, '%2.2f');
%posmin=floor(pos/480000);
%posseg=floor(mod(pos/8000,60));
%postxt=[num2str(posmin) num2str(posseg)];
set(handles.showmatch, 'String', postxt);

% --- Executes on selection change in showmatch.
function showmatch_Callback(hObject, eventdata, handles)
% hObject    handle to showmatch (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns showmatch contents as
cell array
%         contents{get(hObject,'Value')} returns selected item from
showmatch
global P Fs pos
inf=get(hObject, 'Value');
gos=get(hObject, 'String');

%reproduce fragmento de AP donde se encuentra una coincidencia de AS
switch inf
case inf
    wavplay(P(pos(inf)-16000:pos(inf)+32000), Fs);
end
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function showmatch_CreateFcn(hObject, eventdata, handles)
% hObject    handle to showmatch (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

```

```
% Hint: listbox controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%cierra ventana resultados
delete(resultados)

% --- Executes on button press in AbrRep.
function AbrRep_Callback(hObject, eventdata, handles)
% hObject    handle to AbrRep (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
winopen Datos.xls
```