

1217 (IF)

OCT 2013

UNIVERSIDAD NACIONAL DEL CALLAO
FACULTAD DE INGENIERÍA INDUSTRIAL Y DE
SISTEMAS

R	UNIVERSIDAD NACIONAL DEL CALLAO
E	VICE-RECTORADO DE INVESTIGACIÓN
C	362
I	14 OCT 2013
E	
I	HORA: 12.55 HR
D	
O	FIRMA: <i>[Signature]</i>

DETERMINACION DE LA DIRECCION DEL ORIGEN DEL SONIDO
PARA SISTEMAS DE ALERTA APLICANDO REDES
NEURONALES ARTIFICIALES

SALLY KARINA TORRES ALVARADO
(RESOLUCIÓN RECTORAL N° 1072-2011-R)
(01 Octubre 2011 – 30 Setiembre 2013)

CALLAO 2013



APÉNDICE F. Código fuente de aprendizaje de la red neuronal artificial (fuente: Elaboración propia)	20
APÉNDICE G. Funcionamiento del prototipo. (fuente: Elaboración propia)	21



1. RESUMEN

Uno de los mecanismos efectivos para evitar accidentes con que contamos las personas es la capacidad para detectar la dirección el origen de sonidos que nos indican la proximidad de situaciones riesgosas, caída de objetos, desplazamiento de objetos, alertas sonoras, gritos etc.

Este mecanismo puede ser aplicado de manera eficaz en sistemas artificiales de alerta para evitar accidentes, alertar sobre posibles riesgos o aplicaciones de otro tipo que requieran esta característica.

Para determinar la dirección de origen del sonido es necesario capturar el sonido a través de dos micrófonos (APENDICE A), en formato WAV (APENDICE B), estos dos sonidos capturados por los micrófonos llegan en diferentes tiempos por la velocidad de difusión de sonido en el aire y esta diferencia se puede usar para determinar la dirección de origen del sonido.

Para no depender de parámetros de los micrófonos y distancia entre ellos y demás parámetros físicos en la presente investigación se resolvió el problema utilizando redes neuronales artificiales, para lo cual se creó una red neuronal artificial que tiene como entrada los dos sonidos capturados por los micrófonos.

La implementación del prototipo con la propuesta de solución, para realizar las pruebas se desarrolló en Visual C++ lográndose probar que si es posible la determinación de la dirección del origen del sonido.

2. INTRODUCCIÓN

La determinación de la dirección de origen del sonido es una de las facultades de las personas que nos sirve para ubicarnos en el medio y nos alerta sobre posibles peligros.

Generalmente el enfoque para resolver este problema consiste en hacer un pre - procesamiento frecuencial de las señales de sonido utilizando métodos matemáticos sin embargo no logran la efectividad que logra una persona, que lo resuelve con las redes neuronales biológicas, por lo que en el presente proyecto se busca crear un modelo con una red neuronal que logre determinar con mayor eficiencia, capturando los sonidos con una arquitectura de hardware similar en cuanto a la ubicación de los micrófonos que sustituyen al sistema auditivo.

Considerando estas particularidades, el problema de la investigación es:

¿Es posible que un sistema pueda alertar sobre riesgos determinando la dirección de origen del sonido aplicando redes neuronales artificiales?

Planteándose como objetivo de determinar la dirección de origen del sonido mediante redes neuronales artificiales para aplicarlo en sistemas de alerta de riesgos que generan sonidos.

Para lo cual se:

- a. Captura sonidos de muestra con micrófonos mediante una computadora.
- b. Estudio de las características de los sonidos capturados por lo diversos micrófonos.

c. Determinación de la dirección de origen del sonido mediante las redes neuronales.

Cabe destacar que la presente investigación se justifica debido que la determinación de la dirección del sonido para las personas es muy importante porque permite entre otras cosas evitar y determinar la presencia de riesgos que generan sonidos, y con esta facultad pueden evitarlos a tiempo, esta cualidad se podría dotar a maquinas para que alerten sobre la presencia de sonidos no comunes que podrían representar algún riesgo para las personas en un lugar físico. En este contexto la presente investigación hace un gran aporte tecnológico permitiendo implementar sistemas de alerta para peligros que generan sonidos. De esta forma la presente investigación tiene importancia y queda justificada.

3. MARCO TEÓRICO

La posición física de los oídos y la forma de las orejas permiten a las personas determinar desde donde se originan los sonidos en el espacio (BEAR 2004), utilizando el sonido que ingresa a los oídos y transformado a una representación frecuencial que ingresa a una red neuronal biológica.

Basándonos en el sistema auditivo biológico, para determinar la dirección de origen del sonido es necesario capturar el sonido mediante dos micrófonos conectados a una computadora y ubicados espacialmente de manera semejante a como están dispuesto los oídos de las personas y guardarlo como archivo en un formato que sea fácil de manipular para lo cual se usa WAV (APÉNDICE A).

Los sonidos capturados por los dos micrófonos tienen un ligero retraso (APENDICE E) entre ellos por la velocidad de difusión del sonido, retraso que se puede utilizar para obtener la dirección de origen del sonido.

Para determinar la dirección de origen del sonido se aplica redes neuronales artificiales (JANG 2007) de perceptrones que permiten clasificar fácilmente datos con características semejantes.

El aprendizaje de la red neuronal consiste en ajustar los pesos de las entradas de cada una de las neuronas, aplicando un método de backpropagation (JANG 2007) modificado, el cual se basa en un método heurístico computacional, que modifica los pesos que influyen en la salida correspondiente.

Para obtener los datos del sonido digital se utiliza el formato WAV (APENDICE B) con los cuales podemos graficar la señal de los sonidos para realizar los estudios de las señales y generar los datos para la red neuronal artificial.

El funcionamiento del sistema consiste en realizar la captura de los sonidos mediante los dos micrófonos, obtener los datos digitalizados de los sonidos y ponerlos a la entrada de la red neural artificial, y realizar el algoritmo de cálculo de la red, el cual consiste en calcular la sumatoria de los productos de las entradas de cada neurona, pasar por la función de activación y propagar el resultado hacia delante hasta llegar a la capa de salida. Este algoritmo esta implementado en Visual C++. El tiempo de obtención del resultado siempre es el mismo y no es afectado por el tipo de datos que se ingresen al sistema.

El sonido que se captura con los dos micrófonos al mismo tiempo da dos señales del mismo sonido retrasadas entre sí y esta característica permite a la red neuronal artificial determinar la dirección de origen del sonido.

4. MATERIALES Y METODOS

4.1. El Universo.- El presente proyecto es aplicable a todas las organizaciones estatales y privadas donde se necesiten sistemas de alerta.

4.2. Técnicas de recopilación de datos.- La presente investigación es de tipo tecnológica – experimental, en la cual se utilizo las siguientes técnicas de recopilación de datos:

- Exploración conceptual y experimental para determinar la mejor herramienta para capturar los sonidos y guardarlo en forma de archivo de sonido.
- Exploración conceptual y experimental para determinar la disposición espacial de los micrófonos para capturar los sonidos.
- Exploración conceptual y experimental para determinar las técnicas de procesamiento del sonido y redes neuronales artificiales para plantear la posible solución al problema.
- Exploración conceptual y experimental para determinar las herramientas tecnológicas de programación para implementar la posible solución al problema planteado en la presente investigación.

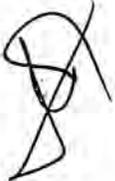


- Para la contrastación de la hipótesis se utiliza resultados de la implementación experimental de un prototipo que permite verificar la determinación de la dirección del origen del sonido.

4.3. Técnicas estadísticas.- La presente investigación no requiere la aplicación de técnicas estadísticas ya que la determinación de origen de sonido no es un proceso estocástico, más bien es determinista y solo depende del funcionamiento del prototipo experimental.

Cabe destacar que ninguna de las variables influyentes en la presente investigación, tienen un comportamiento estocástico, excluyendo de esta forma la aplicación de técnicas estadísticas.

4.4. Materiales.- Para la realización del experimento es necesario los siguientes materiales:

- 
- Computadora Core i5, Disco duro de 500 GB; RAM 8 GB; monitor; teclado; o cualquier equipo superior.
 - Sistema operativo Microsoft Windows.
 - Lenguaje de programación Microsoft Visual C++ 2010.

4.5. Procedimiento.- Para ejecutar el experimento se debe:

- Copiar el código fuente en la computadora con Visual C++ 2010 en el disco C desde la carpeta /sonido del CD.
- Ejecutar el programa con Visual C++ 2010.

- Verificar la determinación del origen del sonido presionando el botón inicio del programa y generando sonido expresando la letra "a" (APÉNDICE D).

5. RESULTADOS

Utilizando los dos micrófonos instalados espacialmente en forma semejante al os oídos humanos (APENDICE A) conectados a una computadora se captura dos muestras de sonido simultáneamente en formato WAV (APENDICE B).

Utilizando el programa en Visual C++ 2010 (APÉNDICE C) que lee los datos de los archivos en formato WAV (APÉNDICE B) se construyo las imágenes de las señales de sonido (APÉNDICE D) capturadas en los archivos WAV.

Las imágenes del sonido capturado por los dos micrófonos sobrepuestas en un mismo sistema de coordenadas (APENDICE E) nos permito ver que el micrófono más distante al origen del sonido captura un sonido ligeramente retrasado con respecto al sonido capturado por el otro micrófono más cercano a origen, esto se puede apreciar con mayor precisión en la imagen de los dos sonidos sobrepuestos representados por su frecuencia (APENDICE E), lo que es físicamente explicable por la velocidad de difusión del sonido en el aire. Utilizando este retraso entre las imágenes de los dos sonidos aplicando una red neuronal de perceptrones (JANG 1997) nos permite identificar la dirección del origen del sonido.

Utilizando el prototipo de programa en Visual C++ 2010 (APÉNDICE F) se realizo el proceso de aprendizaje de la red neuronal artificial para las imágenes generadas

por sonidos capturados en diversas posiciones. Y se comprobó el funcionamiento con el prototipo en Visual C++ 2010 (APÉNDICE G).

De esta forma la presente investigación ha permitido comprobar que utilizando redes neuronales artificiales es posible determinar la dirección del origen del sonido en tiempos muy cortos y puede ser aplicado para sistemas de alerta.

El resultado de la presente investigación genera la experiencia científica y tecnológica en la implementación de sistemas que determinan la dirección de origen del sonido con redes neuronales artificiales en un lenguaje de programación común como es el Visual C++ 2010, que puede ser aplicado para sistemas de alerta.



Los resultados obtenidos durante la presente investigación demuestran que las redes neuronales artificiales permiten determinar la dirección de origen del sonido para aplicar en sistemas de alerta y de esta manera queda demostrada la hipótesis planteada en la presente investigación.

6. DISCUSIÓN

Si bien otros autores proponen soluciones al presente problema, en la presente investigación se caracteriza por lograr desarrollar el método y nuestra propia tecnología lo que nos permite utilizar sin reconocer derechos de patentes de otros investigadores de otros países.

En este trabajo de investigación se propuso resolver el problema de determinación de la dirección de origen del sonido utilizando redes neuronales artificiales, lo cual con un hardware adecuado, disponiendo los micrófonos espacialmente como los

oídos humanos, se logro desarrollar y realizar las pruebas del método para determinar la dirección de origen del sonido en lenguaje C++ lo que puede ser también implementado en otros lenguajes de programación.

De esta forma en el presente trabajo de investigación se cumplido con el objetivo de determinar la dirección de origen del sonido aplicando redes neuronales artificiales con herramientas de programación de uso común como es el C++ y puede ser programado en cualquier otro lenguaje de programación que tenga la posibilidad manejo de archivos y matrices, de esta forma no es necesario de la adquisición de costosas librerías desarrollada para tales fines, logrando el desarrollo tecnológico en esta área.

Otro de los grandes beneficios es que las redes neuronales requieren un bajo procesamiento computacional, lo que permite utilizar para este tipo de sistema, computadoras personales de bajo costo, haciendo posible la utilización de esta alta tecnología a menor costo.

7. REFERENCIAS

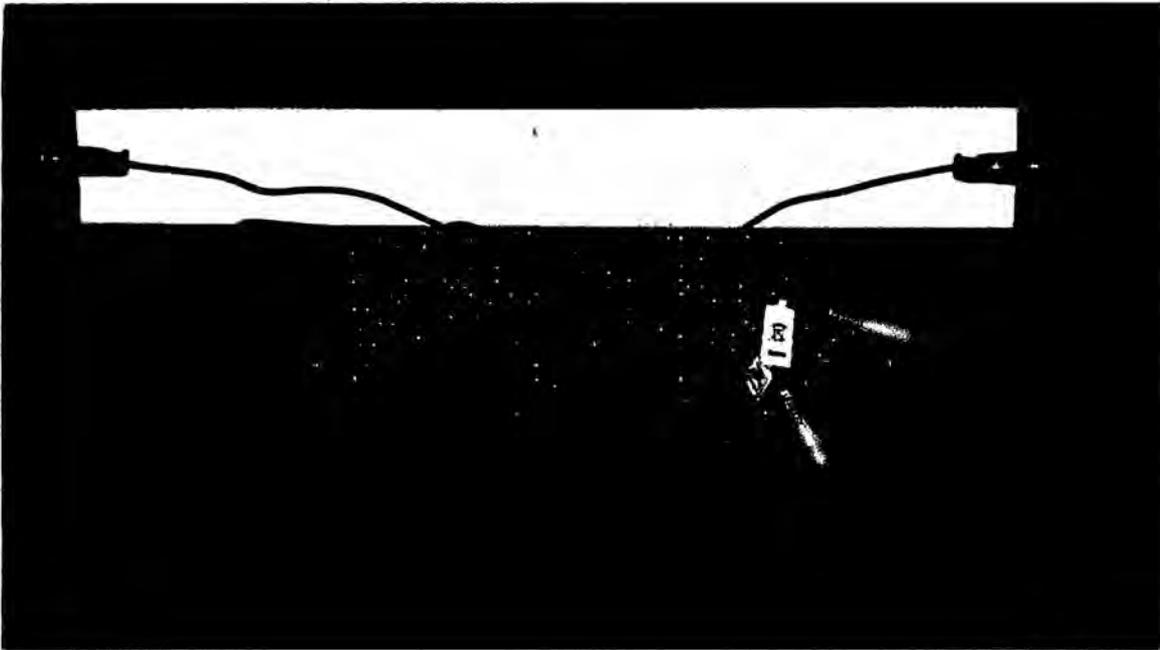
- 7.1. Andrzej C. Automatic identification of sound source position employing neural networks and rough sets. Pattern Recognition Letters 24. Pg. 921–933. 2003.
- 7.2. BEAR M. F., CONNORS B. W., PARADISO M. A., Neurociencia explorando el cerebro. Barcelona: MASSON S.A. Primera edición. 2004.

- 7.3. Fayin, L. Wechsler, H. Open Set Face Recognition Using Transduction. IEEE Transactions on Pattern Analysis and Machine Intelligence, November 2005, v.27 n.11, p.1686-1697.
- 7.4. JANG, J. -S. R. SUN, C. -T. MIZUTAMI, E. Neuro – Fuzzy and Soft Computing A Computational Approach to Learning and Machine Intelligence, Upper Saddle River: Prentice Hall INC., 1^{ra}, 1997.
- 7.5. Lasecki, J. Kostek B. Czyzewski A. Neural Network-Based Spatial Filtration of Sound. AES E-Library AES Convention: 106. Paper Number: 4918. Mayo 1999.
- 7.6. Juha B. Matti K. Modelling of human directional and spatial hearing using neural networks. IEEE Computer Society Washington, DC, USA ©1993.
- 7.7. Oropeza Rodriguez J. Algoritmos y Métodos para el Reconocimiento de Voz en Español Mediante Sílabas. Mexico, Computación y Sistemas, 2006, Vol. 9 Núm. 3, pp. 270-286.
- 7.8. Rowley, H. Baluja, S. Kanade T. Neural Network-Based Face Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, January, 1998, Vol. 20, No. 1, pp. 23-38.
- 7.9. Torres Alvarado S. Reconocimiento facial para identificación de personas mediante redes neuronales. Informe técnico Res.: 372.07-RR – UNAC. Callao. Setiembre 2008.

- 7.10. Torres Alvarado S. Auto-ajuste de la estructura de la red neuronal artificial para un sistema de reconocimiento facial aplicando algoritmos heurísticos. Informe técnico Res.: 1127.08-R – UNAC. Callao. Marzo 2010.
- 7.11. Torres Alvarado S. Reconocimiento de voz para convertir el discurso hablado en texto aplicando redes neuronales artificiales. Informe técnico Res.: 416-2010-R – UNAC. Callao. Setiembre 2011.

8. APÉNDICE

APÉNDICE A. Micrófonos instalados espacialmente de forma semejante a los oídos humanos. (Elaboración propia).



APÉNDICE B. Especificación de formato wave.

Descripción.- El formato wave es nativo de Windows para almacenar datos de audio y es usado por la mayoría de aplicaciones de audio.

Estructura del archivo wave.- Los archivos wave usan el estándar RIFF, que consisten en almacenamiento de datos de audio en porciones separadas que contienen su propia cabecera y datos. La cabecera de la porción contiene el tipo y el tamaño de datos.

Cabecera del archivo wave.- Los primeros 8 bytes contiene el identificador de porción y el tamaño de los datos que es igual a el tamaño de archivo menos 8 bytes.



Offset	Size	Description	Value
0x00	4	Chunk ID	"RIFF" (0x52494646)
0x04	4	Chunk Data Size	(file size) - 8
0x08	4	RIFF Type	"WAVE" (0x57415645)
0x10	Wave chunks		

Formato de la porción.- El formato contiene información sobre como esta almacenado los datos y como debe ser reproducido incluyendo el formato de compresión, numero de canales, periodo de muestreo, bits por muestreo y otros.

Offset	Size	Description	Value
0x00	4	Chunk ID	"fmt " (0x666D7420)

0x04	4	Chunk Data Size	16 + extra format bytes
0x08	2	Compression code	1 - 65,535
0x0a	2	Number of channels	1 - 65,535
0x0c	4	Sample rate	1 - 0xFFFFFFFF
0x10	4	Average bytes per second	1 - 0xFFFFFFFF
0x14	2	Block align	1 - 65,535
0x16	2	Significant bits per sample	2 - 65,535
0x18	2	Extra format bytes	0 - 65,535
0x1a	Extra format bytes *		

Códigos de compresión

Code	Description
0 (0x0000)	Unknown
1 (0x0001)	PCM/uncompressed
2 (0x0002)	Microsoft ADPCM
6 (0x0006)	ITU G.711 a-law
7 (0x0007)	ITU G.711 μ -law

17 (0x0011)	IMA ADPCM
20 (0x0016)	ITU G.723 ADPCM (Yamaha)
49 (0x0031)	GSM 6.10
64 (0x0040)	ITU G.721 ADPCM
80 (0x0050)	MPEG
65,536 (0xFFFF)	Experimental

Datos de audio.- Contiene los datos de audio de cada muestra que debe ser decodificada de acuerdo a método de compresión especificado en la cabecera.

Offset	Length	Type	Description	Value
0x00	4	char[4]	chunk ID	"data" (0x64617461)
0x04	4	dword	chunk size	depends on sample length and compression
0x08	sample data			

APÉNDICE C. Programa para construir imágenes de los sonidos almacenados

en archivos en formato WAV (Fuente: Elaboración propia)

```
x=fopen("o.wav","rb");
y=fopen("data.txt","w");
if(!x){
    exit(0);
}
else
{
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "ChunkID : %x\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "ChunkSize : %d\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "Format : %x\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "Subchunk1ID : %x\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "Subchunk1Size : %u\n", son);
    fread(&faudio, sizeof(unsigned short int), 1, x);
    fprintf(y, "AudioFormat : %u\n", faudio);
    fread(&canales, sizeof(unsigned short int), 1, x);
    fprintf(y, "NumChannels : %u\n", canales);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "SampleRate : %u\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "ByteRate : %u\n", son);
    fread(&canales, sizeof(unsigned short int), 1, x);
    fprintf(y, "BlockAlign : %u\n", canales);
    fread(&canales, sizeof(unsigned short int), 1, x);
    fprintf(y, "BitsPerSample : %u\n", canales);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "Subchunk2ID : %u\n", son);
    fread(&son, sizeof(unsigned long), 1, x);
    fprintf(y, "Subchunk2Size : %d\n", son);
    fprintf(y, "Datos : \n");
    datson=0;
    datsona=0;
    muestra=0;
    muestraa=0;
    cmuestra=0;
    do{
        cmuestra++;
        fread(&datson, sizeof(unsigned short int), 1, x);
        if ((cmuestra>=inigraf) & (cmuestra<=finigraf)){
            muestra++;
            pDC->MoveTo(xc1+exe*muestraa,yc1-eye*datsona);
            pDC->LineTo(xc1+exe*muestra,yc1-eye*datson);
            muestraa=muestra;
            datsona=datson;
        }
        fprintf(y, "Datos : %d\n", datson);
    }while (!feof(x));

fclose(x);
fclose(y);
}
```

APÉNDICE D. Imágenes de los sonidos capturados mediante los dos micrófonos

(Fuente: Elaboración propia)

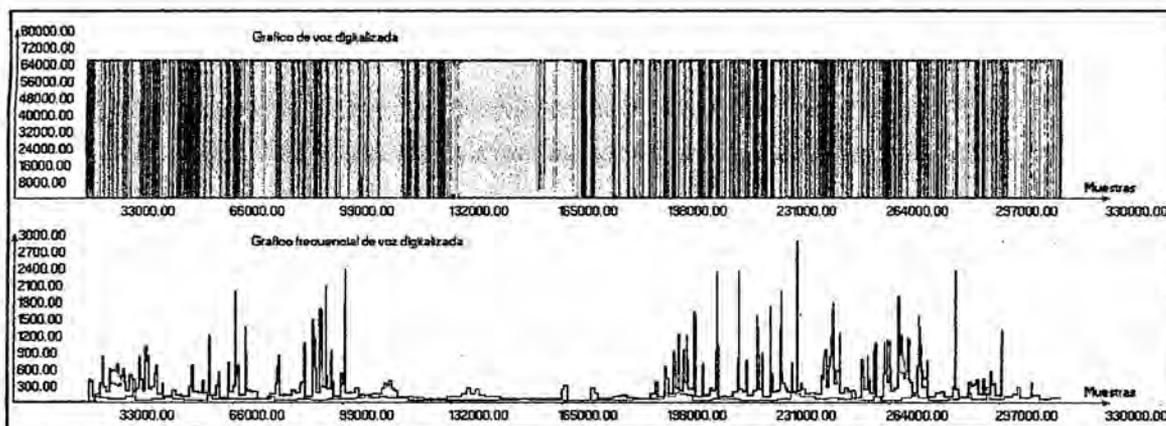


Figura N°- 1.- Señales con la fuente de sonido a la derecha.

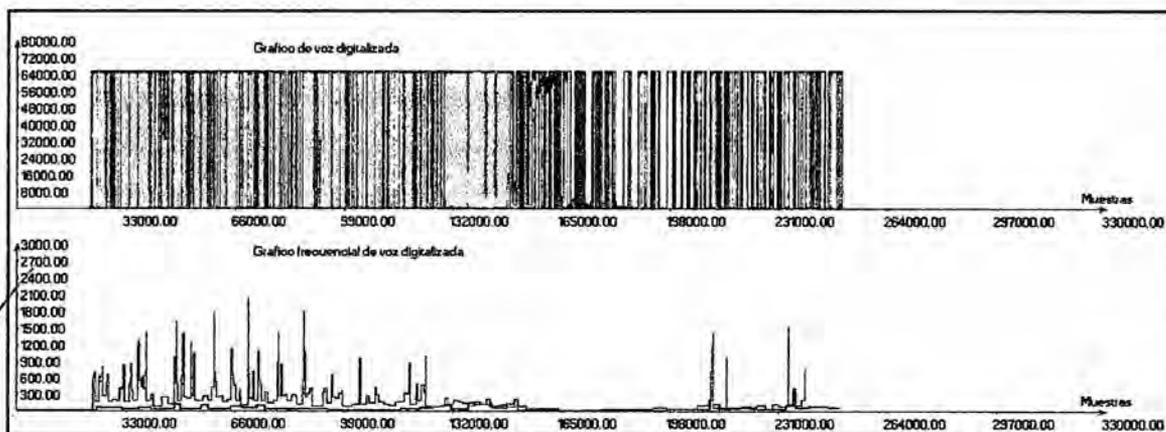
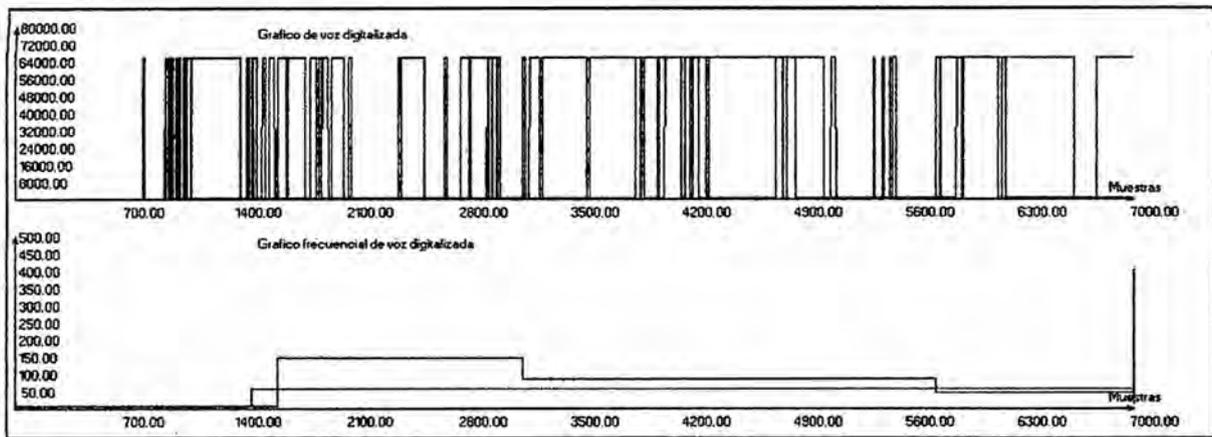


Figura N°- 2.- Señales con la fuente de sonido a la izquierda.

**APÉNDICE E. Imágenes de los sonidos capturados por los dos micrófonos
sobrepuestos en un solo sistema de coordenadas ampliados en
el inicio de la captura (Fuente: Elaboración propia)**



APÉNDICE F. Código fuente de aprendizaje de la red neuronal artificial (Fuente: Elaboración propia)

```

/* Definición de la estructura de la red*/

sred[0]=4;
sred[1]=20;
sred[2]=40;
sred[3]=20;
sred[4]=1;

int ir=0;
for (int j=1;j<=sred[0];j++){
    if (j==1){
        for (int k=0;k<sred[j];k++){
            ma[ir].m=(int)ancho*alto;
            ma[ir].capa=j;
            for (int p=0;p<ma[ir].m;p++){
                ma[ir].peso[p]=rand()*0.000001;
            }
            ma[ir].k=1;
            ma[ir].u=6000;
            ir++;
        }
    }
    else{
        for (int k=0;k<sred[j];k++){
            ma[ir].m=sred[j-1];
            ma[ir].capa=j;
            for (int p=0;p<ma[ir].m;p++){
                ma[ir].peso[p]=rand()*0.000001;
            }
            ma[ir].k=1;
            ma[ir].u=6000;
            ir++;
        }
    }
}

int nred=0;
for (int j=1;j<=sred[0];j++){
    nred=nred+sred[j];
}

fprintf(r,"\n\nEstrucutra de la Red\n\n");
fprintf(r,"Numero de neuronas de la red=%d\n",nred);
for (int j=0;j<nred;j++){
    fprintf(r,"rna[%d].m=%d : rna[%d].capa=%d : rna[%d].k=%f\n",j,ma[j].m,j,ma[j].capa,j,ma[j].k);
}

/* Aprendizaje */

/* Calcular la red */

float res=CalculoRed(nred);

fprintf(r,"\nResultado=%f\n",res);
fprintf(r,"\nm=%d\n",ma[0].m);
fprintf(r,"\nu=%f\n",rna[0].u);
int ne=0;
float AP=0.005;
float ka=0.000001;
float na=0;

```

```

float error=0;
float res1=0;
unsigned int ni=0;
unsigned int nia=0;
float exe=dx/fx;
float eye=dy/pmax;
fprintf(r,"nexe=%fn",exe);
fprintf(r,"neye=%fn",eye);
pDC->MoveTo(xc1,yc1-eye*ma[nred-1].u);
pDC->LineTo(xc1+exe*fx,yc1-eye*rna[nred-1].u);
pOldPen=pDC->SelectObject(&myPen1);

do{
    for (int j=0;j<nred;j++){
        res=CalculoRed(nred);
        if (res<rna[nred-1].u){
            error=ma[nred-1].u+100-res;
            if (ma[j].capa==1){
                for (ne=0;ne<rna[j].m;ne++){
                    if (e[ne]>0){
                        ma[j].peso[ne]=ma[j].peso[ne]+error*ka*e[ne];
                    }
                    ni++;
                }
            }
            else{
                for (ne=0;ne<rna[j].m;ne++){
                    ma[j].peso[ne]=ma[j].peso[ne]+error*ka;
                    ni++;
                }
            }
        }
        pDC->MoveTo(xc1+exe*nia,yc1-eye*na);
        pDC->LineTo(xc1+exe*ni,yc1-eye*res);
        nia=ni;
        na=res;
    }
}while (res<rna[nred-1].u);

for (int j=0;j<nred;j++){
    fprintf(r,"ma[%d].m=%d . ma[%d].capa=%d : ma[%d].k=%fn",j,ma[j].m,j,ma[j].capa,j,ma[j].k);
    for (int pp=0;pp<ma[j].m;pp++){
        fprintf(r,"%f - ",ma[j].peso[pp]);
    }
    fprintf(r,"n\n");
}
pDC->TextOut(600,300,"Fin de aprendizaje");
fclose(r);

```

APÉNDICE G. Funcionamiento del prototipo. (Fuente: Elaboración propia).

