

## Reconocimiento de patrones de gestura empleando redes neuronales, buscando herramientas de inclusión a personas con amputaciones de mano o antebrazo

### Recognition of management patterns using neural networks, looking for inclusion tools for people with hand or forearm amputations

RIVERA-CENICEROS, Omar Fabián†\* & DÍAZ-NUÑEZ, Cintya Yulem

*Universidad Politécnica de Durango, Ingeniería en Telemática. Carretera Durango-México K.M. 9.5*

ID 1<sup>er</sup> Autor: *Omar Fabian, Rivera-Ceniceros* / ORC ID: 0000-0002-4382-5737, Researcher ID Thomson: S-4656-2018, CVU CONACYT ID: 352280

ID 1<sup>er</sup> Coautor: *Cintya Yulem, Díaz-Nuñez*

Recibido 4 de Octubre, 2018; Aceptado 6 de Diciembre, 2018

#### Resumen

El diseño del algoritmo de contrapropagación se presenta a través del procesamiento para crear una red neuronal, a través de la cual se reconocen una serie de manos gesturas utilizando la pulsera myo, que reconoce las señales musculares del antebrazo; Se utiliza como sistema de automatización del hogar para ayudar a las personas que sufren de paraplejia, sistema de rehabilitación o sistema de motricidad fina o amputación.

**Sistema, Red Neuronal, Señales musculares, Paraplejia, Amputación, Tecnología asistiva, MYO®**

#### Abstract

Backpropagation algorithm design is presented through Processing for creating a neural network, through which a series of gesturas hands are recognized using the myo bracelet, which recognizes the forearm muscle signals; It is used as a home automation system to help people suffering from paraplegia, rehabilitation system or fine motor system or amputation.

**System, Neural Network, Muscle signals, Paraplegia, Amputation, Assistive technology, MYO®.**

**Citación:** RIVERA-CENICEROS, Omar Fabián & DÍAZ-NUÑEZ, Cintya Yulem. Reconocimiento de patrones de gestura empleando redes neuronales, buscando herramientas de inclusión a personas con amputaciones de mano o antebrazo. Revista de Innovación Sistemática 2018. 2-8:8-17

\*Correspondencia al Autor (Correo Electrónico: Omar.rivera@unipolidgo.edu.mx)

† Investigador contribuyendo como primer autor.

## Introducción

El hombre se ha caracterizado siempre por su búsqueda constante de nuevas vías para mejorar sus condiciones de vida. Estos esfuerzos le han servido para reducir el trabajo en aquellas operaciones en las que la fuerza juega un papel primordial. Los progresos obtenidos han permitido dirigir estos esfuerzos a otros campos, como por ejemplo, a la construcción de máquinas calculadoras que ayuden a resolver de forma automática y rápida determinadas operaciones que resultan tediosas cuando se realizan a mano [1].

En inteligencia artificial (IA), las redes neuronales son más que otra forma de emular ciertas características propias de los humanos, como la capacidad de memorizar y de asociar hechos. Si se examinan con atención aquellos problemas que no pueden expresarse a través de un algoritmo, se observará que todos ellos tienen una característica en común: la experiencia. El hombre es capaz de resolver estas situaciones acudiendo a la experiencia acumulada. Así, parece claro que una forma de aproximarse al problema consista en la construcción de sistemas que sean capaces de reproducir esta característica humana.

Una red neuronal es "un nuevo sistema para el tratamiento de la información, cuya unidad básica de procesamiento está inspirada en la célula fundamental del sistema nervioso humano: la neurona" [2].

El funcionamiento de la Red Backpropagation (BPN) consiste en el aprendizaje de un conjunto predefinido de pares de entradas-salidas dados como ejemplo: primero se aplica un patrón de entrada como estímulo para la primera capa de las neuronas de la red, se va propagando a través de todas las capas superiores hasta generar una salida, se compara el resultado en las neuronas de salida con la salida que se desea obtener y se calcula un valor de error para cada neurona de salida. A continuación, estos errores se transmiten hacia atrás, partiendo de la capa de salida hacia todas las neuronas de la capa intermedia que contribuyan directamente a la salida, recibiendo de error aproximado a la neurona intermedia a la salida original.

Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido un error que describa su aportación relativa al error total. Basándose en el valor del error recibido, se reajustan los pesos de conexión de cada neurona, de manera que en la siguiente vez que se presente el mismo patrón, la más salida esté cercana a la deseada [3].

Gracias a las Redes Neuronales se pueden sistemas que ayuden a distintas problemáticas, como en este caso de rehabilitación en el motor fino, e incluso en parálisis siendo desarrollado como un sistema doméstico.

Consultando las estadísticas se puede observar que más de 1,000 millones de personas sufren de algún tipo de discapacidad, lo que constituye aproximadamente el 15% de la población mundial, y de ellas una quinta parte se ve enfrentada a grandes dificultades en su vida cotidiana, según reveló un informe conjunto de la Organización Mundial de la Salud (OMS) y el Banco Mundial (BM) [4]

El informe, que es el primero que se realiza de manera global sobre este problema en 40 años, destaca que muy pocos países cuentan con mecanismos adecuados para responder a las necesidades de las personas con discapacidad.

Los accidentes laborales que son lesiones físicas que el trabajador puede sufrir como consecuencia de la realización de las actividades propias de su trabajo y generen alguna discapacidad. Según las cifras, se indica que en México aumentó un 6,9%. En cifras absolutas, el número total se eleva hasta los 400.000 accidentes laborales, teniendo 1,100 accidentes laborales diarios [6].

En México existen 5 millones 739,720 personas con discapacidad. El 58.3% de ellos se encuentran inhabilitado para caminar o moverse; en segundo lugar, el 27.2% de ellos presenta alguna dificultad para ver y el 12.1% tiene problemas en su capacidad para escuchar, según el censo de población y vivienda 2010 realizado por el Instituto Nacional de Estadística y Geografía.

Las causas de discapacidad de esta población están relacionadas con alguna enfermedad o accidente en el 39.4% de los casos,. Entre 110 y 190 millones de los discapacitados se enfrentan a barreras que van desde el estigma y la discriminación hasta la ausencia de adecuados servicios de atención sanitaria y rehabilitación, así como sistemas de transporte o edificios inaccesibles y falta de oportunidad laboral sobre todo en empresas de manufactura de productos donde trabajadores productivos son cesados o indemnizados donde la Ley Federal del Trabajo precisa, por ejemplo, que por la pérdida total de la mano el afectado recibirá un porcentaje de incapacidad indemnizable, de acuerdo a su salario, del 65 al 75%; por el pie, 50 a 55%; brazo 70 a 80%; por los dos testículos, 40 al 100%, considerando la edad del paciente [5].

### Planteamiento del Problema

Desarrollar una herramienta aplicando las nuevas tecnologías de la información, comunicación y electrónica para brindar apoyo mediante tecnología asistiva a personas con discapacidades motrices por enfermedad o accidente laboral, amputaciones, etcétera, tomando en consideración que personas jóvenes o laboralmente productivas puedan seguir ofreciendo su profesión o puedan ser empleadas en la industria, fomentando su inclusión en el ámbito laboral [6].

### Justificación

Existen distintos sistemas de ayuda a personas con discapacidad motriz en cuanto a su desplazamiento, pero ¿Qué hay de la problemática que surge al tratar de incorporarlos al ámbito laboral en un sector específico? Con el proyecto se pretende incorporar al trabajador aun productivo, con la posibilidad de controlar algunos sistemas eléctricos, electromecánicos, etc. Con el desarrollo de un proyecto capaz de interactuar con otro sistema de control que facilitará la manipulación de distintos aparatos mediante una combinación de movimientos, haciendo que las personas que sufran de alguna disfuncionalidad motriz puedan ser autosuficientes y funcionales en sectores industriales como el manufacturero siempre y cuando el trabajo que requieran hacer no necesite hacer uso preciso de las manos.

Con el desarrollo de este algoritmo, se puede incluso crear un sistema de rehabilitación para pacientes que sufran algún problema en el sistema motor fino, implementando un plan de ejercicios a seguir, los cuales deberán ser ejecutados como medida de rehabilitación.

### Objetivo

Diseñar un algoritmo que sea capaz de reconocer las gesturas de la mano a través de una red neuronal, haciendo uso del brazalete MYO como sensor para crear una herramienta asistiva que permita a trabajadores con discapacidad mantenerse productivos en el ámbito laboral, preferentemente en la industria manufacturera.

### Diseño del Sistema

#### Diseño

El diseño del sistema permite la descripción detallada de los elementos y etapas que conforman el proyecto, tomando en cuenta las características de mayor importancia dentro del mismo, basándose en la manipulación binaria de sistemas de actuadores que no requieran el uso preciso de las manos. Cabe hacer mención que al estar colocados los sensores en el antebrazo, no importa si la mano ha sido amputada, ya que los movimientos son registrados en el musculo extensor y aductor.

#### Etapas de diseño y entrenamiento

Se ha elegido el brazalete Myo como dispositivo para reconocer las gesturas a realizar, ya que cuenta con una serie de movimientos ya establecidos para su manipulación, además de alta precisión al momento de reconocer los movimientos.

El brazalete cuenta con cinco gesturas establecidas, entre las cuales una de ellas de obtener más que solo cuatro resultados, se optó por diseñar una combinación de tres gesturas, para así contar con cinco resultados más, quedando de la siguiente manera:

Gestura 1	Gestura 2	Gestura 3	Resultado
Fist 	Fingers Spread 	Fist 	10
Wave Right 	Wave Left 	Wave Right 	19
Wave Right 	Wave Right 	Fist 	30
Fist 	Fist 	Wave Right 	39
Fingers Spread 	Fingers Spread 	Wave Right 	50
Fingers Spread 	Fist 	Fingers Spread 	59
Wave Left 	Wave Right 	Wave Left 	78
Wave Left 	Wave Left 	Fist 	80
Fist 	Fist 	Wave Left 	89

**Tabla 1** Combinación de gesturas

De esta manera cada resultado se asigna a determinada acción de manera más sencilla. El diseño de la Red Neuronal se realizó a partir de la serie de gesturas, es decir con tres entradas de datos y una sola salida.

Para el entrenamiento de la Red Neuronal se usó el software Matlab, usando la librería neuralnet para la simulación, obtención, las comprobaciones y resultados según el algoritmo construido, para obtener los resultados de una manera eficiente fue necesario solo hacer uso de 5 neuronas que construyeron la red exitosamente.

## Etapas de control IA y de potencia

Para la etapa de control se desarrolló un algoritmo en Processing, haciendo uso de los SDK: firmata, Arduino, myo y serial; gracias a estas librerías el desarrollo y funcionamiento del programa se hace más sencillo, pues con la librería firmata y Arduino es posible enviar los comandos directamente hacia la tarjeta de desarrollo sin la necesidad de programarla previamente y enviar en posterior instancia los datos el un hacia Arduino y de este a la tablilla, es decir el algoritmo completo fue programado en Processing.

Se inició por programar la red neuronal según su entrenamiento basado en el método de la red Backpropagation, enseguida se creó el algoritmo de control, mediante ciclos de IF se validó para al momento de obtener el número final según la tabla de gesturas construidas se encendiera el puerto deseado.

## Etapa Microcontrolada

Para esta etapa fue elegida la tarjeta de desarrollo Arduino Mega ya que cuenta principalmente con una gran cantidad de puertos (52 Puertos), el cual es de gran ayuda para este tipo de sistemas en donde puede ser escalable y así poder agregar más funciones al sistema. Esta etapa es muy importante ya que recibe las órdenes de la etapa de control principal y realiza el control de la tarjeta de potencia para poder cambiar el giro de los motores y pararlos.

## Etapa de potencia

Esta etapa es la que se encarga del encendido de los puertos del Arduino y por consecuente los módulos de relevadores que van conectados al puerto correspondiente, de esta manera el aparato o herramienta eléctrica o electrónica que está conectado al módulo, encenderá al realizar correctamente la serie de gesturas configuradas.

## Pruebas

En esta etapa se realiza la comprobación del funcionamiento del sistema, en el cual se realizan pruebas para determinar si las series de gesturas están siendo detectadas de manera correcta y si el programa está dando el resultado estipulado. Es muy importante esta etapa, para la verificación de la funcionalidad.

**Desarrollo del proyecto**

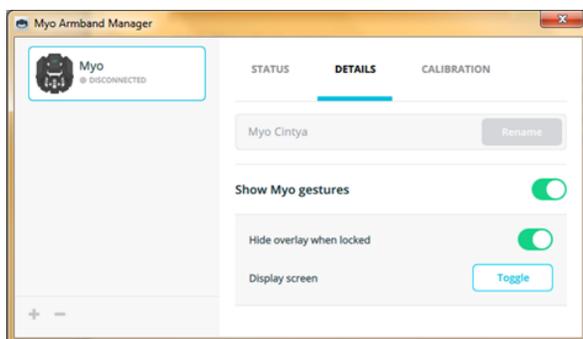
Ahora se explicará acerca de los procesos que fueron necesarios en la realización del sistema y se explicara brevemente los detalles de elaboración de cada parte del sistema.



**Figura 1** Diagrama de bloques del funcionamiento del sistema

**Obtención de Gesturas**

El brazalete Myo obtiene las señales musculares mediante acelerómetros y sensores, este mismo trae incluido un bluetooth, el cual es conectado a la computadora, es cuestión de colocarlo en el antebrazo y a través del software de configuración Myo Connect se crea un perfil para adaptar el brazalete al usuario, de esta manera se comienzan a detectar las gesturas por la computadora, en cuanto a Processing, cuenta con la librería Myo, a través de la cual las gesturas son reconocidas y pueden ser programadas de manera más sencilla con únicamente los nombres de cada gestura.

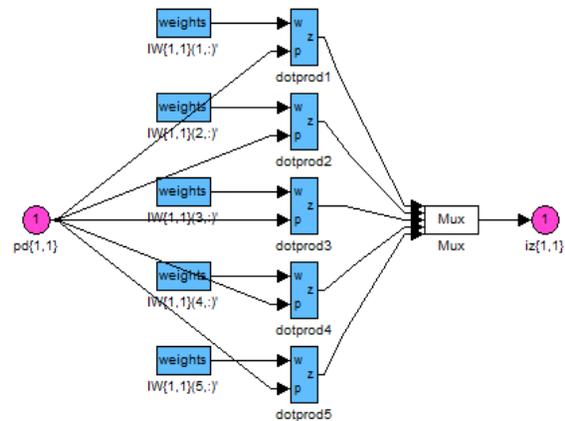


**Figura 2** Interface “Myo Connect”

**Diseño y entrenamiento de Red Neuronal**

La Red Neuronal fue diseñada a partir de la tabla de verdad construida mediante la serie de gesturas a realizar, en la cual los valores son de 1 y 0, enseguida son normalizados según lo pide el método de Backpropagation a 1 y -1 para su entrenamiento.

La Red Neuronal se entrenó a partir de Matlab y la librería neuralnet, donde se ingresaron las salidas deseadas y este arrojó los pesos con los que se debería iniciar y cuantas neuronas serían necesarias para obtener el resultado. A partir de estos resultados y del método Backpropagation se fue construyendo el algoritmo de la Red Neuronal.



**Figura 3** Red Neuronal simulada en Matlab

**Diseño y Programación de Algoritmo de la Red Neuronal**

El algoritmo de la Red Neuronal se construyó a partir de lo siguientes pasos

Paso 1: Declarar los datos según la serie de gesturas en la tabla, así como los pesos de la red, los cuales se obtuvieron a través de MATLAB así como un valor adicional que se indica para los cálculos, en este caso llamado teta.

Para normalizar los datos se usó la siguiente fórmula:

$$f(x) = \frac{(x-dL)*(nH-nL)}{dH-dL} - nL \tag{1}$$

Dónde:

$$nL = -1$$

$$nH = 1$$

$$dH = \text{Rango inferior de } X = 4$$

$$dL = \text{Rango superior de } X = 1$$

*x = Entrada de dato según gestura*

Paso 2: Se aplica la siguiente fórmula a cada uno de las series de gesturas con los datos ya normalizados.

$$Y_i = f(\sum(W_{ij})(X_j) + \theta_i) \quad (2)$$

Dónde:

$W$  = Pesos de la red

$X$   
= Entrada de dato según gestura ya normalizada

Enseguida se calculan los términos de error de cada una de las neuronas, en general se dispone de dos formas de función de salida, selección de la función depende de la forma que se decida representar la salida: si se desea que las neuronas de salida sean binarias, se utiliza la función sigmoideal, en otros casos, la lineal; en este caso se usará la sigmoideal, la cual tiene la siguiente fórmula:

$$f_{x(\text{net}_{jk})} = \frac{1}{1 + e^{-\text{net}_{jk}}} \quad (3)$$

Al tener el resultado de error de cada una de las neuronas se realiza la sumatoria de cada una de ellas multiplicado por los nuevos pesos y el sumando al final el nuevo valor de teta, también obtenidos del entrenamiento basado en Matlab:

$$Y = f(\sum(W_i)(X_j) + \theta) \quad (4)$$

Finalmente calcula el resultado final, mediante el cual se realiza el programa de control; este resultado se obtiene a partir de la siguiente fórmula de desnormalización:

$$f(x) = \frac{(dL - dH)(x) - (nH - dL) + dH * nL}{nL - nH} \quad (5)$$

Dónde:

$$nL = -1$$

$$nH = 1$$

$$dH = \text{Rango inferior de } X = 9$$

$$dL = \text{Rango superior de } X = 1$$

$X$  = Entrada de dato según gestura ya normalizada

A continuación se presenta el código de la Red Neuronal, el cual fue programado mediante el IDE Processing:

Los mostrados a continuación son los pesos con los cuales será iniciada la red neuronal, estos fueron tomados de MATLAB a partir del entrenamiento, así como el valor dado a teta.

```
float[] w1={1.6917643174431292,-
1.6304818882143501,0.47775380225696706};
float[] w2={-2.1536830291165283,-
0.29685226371094053,-
1.7676526247278765};
float[] w3={1.9408282697889678,-
1.414712343904412,0.6897175247779942};
float[] w4={0.29273822477095091,-
1.818513262878142,1.1533870149346594};
float[] w5={1.3810305164926924,0.51001706190511
12,2.2483523806203496};
float[] teta={-
2.3232206835210398,1.1583749671878683,0.8
829327127677028,2.697341148250203,2.2103
953477717493};
```

Aquí se presentan los datos en su forma natural, antes de comenzar a realizar los cálculos.

```
int x1[9]={1,2,2,1,4,4,3,3,1};
int x2[9]={4,3,2,1,4,1,2,3,1};
int x3[9]={1,2,1,2,2,4,3,1,3};
```

Para normalizar los datos presentados como naturales, se usaron las siguientes constantes, las cuales fueron descritas en el apartado de las ecuaciones.

```
float dh=4;
float dl=1;
float nl=-1;
float nh=1;
```

Las siguientes son las constantes utilizadas en las desnormalización:

```
float dhd=9;
float dld=1;
float nld=-1;
float nhd=1;
int des;
```

Se declararon también los pesos para la neurona en la capa 2, obtenidos también de MATLAB.

```
wf={0.38509900215215531,1.2430052876776
403,1.8885969152536324,-
0.99505254977050595,0.41360518246098443}
;
float steta=-0.18866617207016767;
```

En la función llamada “norma” se lleva a cabo la función de normalizar los datos.

```
void norma(){
norma1=(((x1-dl)*(nh-nl))/(dh-dl))+nl;
norma2=(((x2-dl)*(nh-nl))/(dh-dl))+nl;
norma3=(((x3-dl)*(nh-nl))/(dh-dl))+nl;
}
```

Se calculan las salidas de la primera capa.

```
void ye(){
ye1=(w1[0]*norma1)+(w1[1]*norma2)+(w1[2]*norma3)+teta[0];
ye2=(w2[0]*norma1)+(w2[1]*norma2)+(w2[2]*norma3)+teta[1];
ye3=(w3[0]*norma1)+(w3[1]*norma2)+(w3[2]*norma3)+teta[2];
ye4=(w4[0]*norma1)+(w4[1]*norma2)+(w4[2]*norma3)+teta[3];
ye5=(w5[0]*norma1)+(w5[1]*norma2)+(w5[2]*norma3)+teta[4];
}
```

Se le aplica una función sigmoïdal a cada salida.

```
void sigmoïdal(){
sigm1=((exp(ye1))-(exp(-ye1)))/((exp(ye1)+(exp(-ye1))));
sigm2=((exp(ye2))-(exp(-ye2)))/((exp(ye2)+(exp(-ye2))));
sigm3=((exp(ye3))-(exp(-ye3)))/((exp(ye3)+(exp(-ye3))));
sigm4=((exp(ye4))-(exp(-ye4)))/((exp(ye4)+(exp(-ye4))));
sigm5=((exp(ye5))-(exp(-ye5)))/((exp(ye5)+(exp(-ye5))));
}
```

Se calcula la salida final.

```
void neuronaFinal(){
ye1=((wf[0]*sigm1)+(wf[1]*sigm2)+(wf[2]*sigm3)+(wf[3]*sigm4)+(wf[4]*sigm5))+steta;
}
```

Para finalizar, se desnormaliza, para obtener el resultado que será usado.

```
void desnorm(){
des=(int)((((dld-dhd)*(ye1))-
(nhd*dld)+dhd*nld)/(-2))*10;
}
```

### Algoritmo de control de interfaz de potencia

El algoritmo de control al igual que el de la Red Neuronal fue programado en Processing, esto gracias a las librerías usadas, mediante las cuales se puede controlar Arduino desde esta plataforma, solo es necesario declarar las librerías a utilizar y subir un archivo a la placa Arduino, el cual viene como ejemplo en la plataforma con el nombre de StandardFirmata.

El resultado de la Red Neuronal al cual en el código se llamó “des” se compara con la tabla de resultados mediante un sencillo ciclo IF, donde a partir de cada uno de los resultados se le ordena que encienda un puerto del Arduino y por consiguiente encenderá uno de los electrodomésticos con los cuales fue probado.

```
if(des==10){
  Arduino.digitalWrite(20,
  Arduino.LOW);
}else if(des==59){
  Arduino.digitalWrite(20,
  Arduino.HIGH);
}else if(des==19){
  Arduino.digitalWrite(22,
  Arduino.LOW);
}else if(des==78){
  Arduino.digitalWrite(22,
  Arduino.HIGH);
}else if(des==30){
  Arduino.digitalWrite(24,
  Arduino.LOW);
}else if(des==39){
  Arduino.digitalWrite(24,
  Arduino.HIGH);
}else if(des==80){
  Arduino.digitalWrite(26,
  Arduino.LOW);
}else if(des==89){
  Arduino.digitalWrite(26,
  Arduino.HIGH);
}else if(des==50){
  Arduino.digitalWrite(28,
  Arduino.HIGH);
  delay(5000);
}
```

```

Arduino.digitalWrite(28,
Arduino.LOW);
} des=0; }

```

## Resultados

A través de este sistema de reconocimiento de gesturas haciendo uso de Redes Neuronales, se ha logrado mejorar la eficiencia en el uso del control de gesturas MYO, al incrementar su capacidad de reconocimiento. Lo anterior llevo a una poseer una amplia capacidad de control de ON-OFF con la cabida de controlar un suficiente número de aparatos electrodomésticos, herramientas eléctricas, maquinaria eléctrica e incluso se pudiera aplicar a debido a su sencillez a maquinaria electromecánica que se pudiera controlar por medio de un sencillo control ON-OFF como prensas hidráulicas con control eléctrico.

Mediante el brazalete Myo, el realizar una serie movimientos se posibilita el control de los aparatos de una manera muy precisa y sencilla de manejar, es un sistema con grandes beneficios para personas que padecen una discapacidad y que no puedan manipular aparatos como los antes mencionados de igual forma que una persona sin discapacidad pudiera hacerlo.

A continuación se muestra la secuencia de movimientos realizados durante las pruebas al encender y apagar algunas herramientas eléctricas que servirán para ejemplificar el alcance del producto, cabe hacer mención que se realizaron algunas modificaciones físicas en los mismos para poder mantener la facilidad de control ON-OFF sin tener que hacer uso de las manos, de igual manera se tomaron las precauciones básicas de seguridad para realizar el experimento.

## Pruebas de aplicación y uso del sistema

A continuación se muestran ilustraciones que tratan de ejemplificar el uso e interacción del brazalete con diferentes herramientas, a través del algoritmo de IA.



**Figura 4** Secuencia para encender/Apagar el taladro



**Figura 5** Taladro de banco Dremel para pruebas



**Figura 6** Secuencia de control de esmeriladora modificada



Figura 7



Figura 7 Secuencia control ON-OFF de cortadora modificada



Figura 8 Cortadora de inglete acondicionada



Figura 9 Secuencia de control de ventilador



Figura 10 Ventilador en estado OFF después de una secuencia correcta

### Conclusiones

En el sistema para reconocer las gesturas a través de Redes Neuronales ampliando la capacidad el brazalete Myo, empleado como sensor de las señales musculares del brazo para reconocimiento de gesturas, se puede observar que es muy preciso y sencillo de usar. De entrada el brazalete cuenta con una aplicación sencilla que permite crear perfiles individuales y adaptables al usuario. Por lo tanto al agregarle un algoritmo inteligente el sistema será muy adaptable a la hora de incrementar las capacidades del sistema hasta un alto número de funciones. Lo anterior se verá traducido a que la persona con discapacidad motriz, amputación o enfermedad que no le permita realizar alguna actividad por algún largo periodo de tiempo, podrá emplear esta herramienta electrónica para continuar sus actividades, permitiendo la inclusión de este tipo de personas en el ámbito industrial, que como ya se ha investigado, en México las cifras no son muy alentadoras para aquellas personas aún productivas que quieren continuar trabajando en algún sector industrial.

Esta herramienta cuenta con la versatilidad de poder ser adaptada a una amplia cantidad de dispositivos, herramientas y aparatos eléctricos y hasta electromecánicos.

Para aquellos que no desempeñen labores en la industria, como por ejemplo la manufacturera o alguna otra donde se pueda emplear la herramienta, se puede aplicar en electrodomésticos y aparatos eléctricos simples que ayuden a domotizar un hogar, permitiendo al discapacitado motriz una alternativa al control de su casa sin la ayuda de terceros, lo que se puede traducir como una mejora en el estado anímico del paciente.

La tecnología presentada también se puede establecer como una herramienta de tecnología asistiva, ya que siembra otro antecedente de uso como técnica de rehabilitación básica para aquellas personas que presenten discapacidad en la mano, muñeca o rehabilitación del sistema del motor fino al ser factible la creación de planes de ejercicios que sean mostrados mediante una interfaz que sea capaz de ir guardando los progresos del paciente.

## Referencias

Plasticidad y aprendizaje. 2015. Universitat Politècnica de Catalunya pp 135.

Accidentes de trabajo que provocan más discapacidad.2017. Recuperado de: Salud180.com. <http://www.salud180.com/salud-dia-a-dia/accidentes-de-trabajo-que-provocan-mas-discapacidad>

Damián Jorge Matich. 2001. Redes Neuronales: Conceptos Básicos y Aplicaciones. Universidad Tecnológica Nacional – Facultad Regional Rosario. pp 23-25

Discapacidad y salud. 16 de enero de 2018. Organización mundial de la salud. Recuperado de: <http://www.who.int/es/news-room/fact-sheets/detail/disability-and-health>

Amputaciones, un riesgo laboral por el que nadie responde. 2018. El Siglo de Torreón. Recuperado de: <https://www.elsiglodetorreon.com.mx/noticia/293819.amputaciones-un-riesgo-laboral-por-el-que-nadie-responde.html>

Isis Olimpia Gutiérrez. 2017. La inclusión laboral de personas con discapacidad en México. La opinión. Recuperado de: <http://www.e-consulta.com/opinion/2017-04-27/la-inclusion-laboral-de-personas-con-discapacidad-en-mexico>

Marínez M; Ríos, A. 2006. La Tecnología En Rehabilitación: Una Aproximación Conceptual. Revista Ciencias de la Salud, volumen 4, numero 2

Chang, J.; Chang, M.; Lin, J.; Heh, J. Implements a diagnostic intelligent agent for problem solving in instructional systems. Proc. Int. Workshop Advanced Learning Technologies IWALT 2000, pp. 29-30

Imam, I. 2004. Adaptive Applications Of Intelligent Agents. Vol. 1Proc. Ninth Int. Symp. Computers and Communications ISCC 2004, pp. 7-12

Chang, J.; Chang, M.; Lin, J.; Heh, J. Implements a diagnostic intelligent agent for problem solving in instructional systems. Proc. Int. Workshop Advanced

Learning Technologies IWALT 2000, pp. 29-30