

Sistema de movimiento de imagen para manipulación de alto riesgo en almacén industrial usando visión artificial

A. Portugal¹, Undergrade Students Members Ricardo Palma University-Peru.
{anaportugal1711 }@gmail.com

Abstract— This project consists of the teleoperated robotic hand control using artificial vision.

Machine vision is a branch of artificial intelligence, which develops image processing thanks to the movement of the human hand, then emulate movement in robotic hand.

The functionality of the project varies depending on the application to be required, this includes cases where the operator is not physically present at the place required or as needed to perform object manipulation involving the safety of human beings.

Palabras Claves— Visión Artificial, Mano Robótica, Industria, Seguridad.

I. INTRODUCCIÓN

A. Justificación de la necesidad del proyecto

En la industria de la manufactura existen procesos que el hombre no debe de interferir directamente, ya que puede poner en riesgo su salud. Por lo cual se realiza la manipulación de esos objetos de una forma indirecta utilizando un equipo que el hombre pueda controlar; el cual garantiza evitar algún incidente.

La variedad y gravedad de los efectos adversos de las sustancias peligrosas y su creciente empleo en la industria y en otras muchas actividades determinan la importancia del riesgo para las personas y para el medio ambiente. La manipulación de productos dañinos o peligrosos puede dar lugar a accidentes con consecuencias de distinta índole.

El proceso en el que suele tener lugar de mayor número de accidentes es el trasvase, desplazamiento durante el que pueden tener lugar proyecciones, salpicaduras, contactos dérmicos, intoxicaciones y quemaduras por incendios. Se debe de evitar el pasar un líquido de un recipiente a otro de forma manual. Es fundamental que los trabajadores estén plenamente capacitados sobre los riesgos que entrañan este tipo de operaciones y sobre la forma de realizarlas en condiciones de seguridad. Así mismo se deben haber establecido procedimientos escritos que describan como realizar los trasvases y manipulación de los mismos en condiciones de seguridad óptimas.

De acuerdo a los datos estadísticos del mes de enero de 2016, se registraron 2562 notificaciones, de las cuales, el 96,88% corresponde a accidentes de trabajo, el 2,54% a **incidentes peligrosos**, el 0,39% a accidentes mortales y el 0,20% a enfermedades ocupacionales.

En Latinoamérica ocurren 36 accidentes de trabajo por minuto y mueren aproximadamente 300 personas con causas relacionadas a accidentes laborales. Estimándose

en América latina que el costo de los accidentes de trabajo y las enfermedades ocupacionales es entre 3% y 10% del producto bruto interno. Ministerio de Trabajo, lo cual se muestra en la Figura 1.

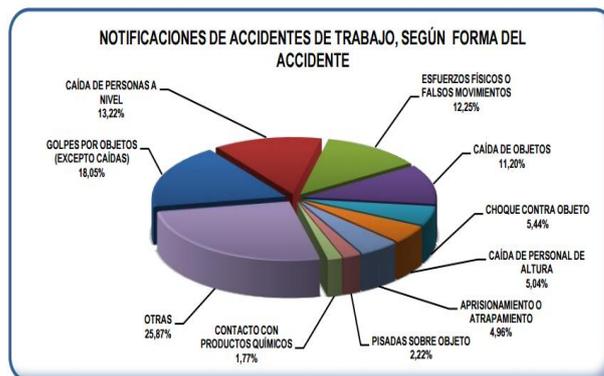


Figura1. Estadística de accidente de trabajo

B. Marco teórico

1. Visión Artificial

La visión artificial es parte de la inteligencia artificial que permite la obtención, procesamiento y análisis de cualquier tipo de información especial obtenida a través de imágenes digitales.

La visión artificial la componen un conjunto de procesos destinados a realizar el análisis de imágenes. Estos procesos son: captación de imágenes, memorización de la información, procesamiento e interpretación de los resultados [2]. Permite la detección automática de la estructura y propiedades de un posible mundo dinámico en 3 dimensiones a partir una o varias imágenes bidimensionales del mundo, tal como se muestra en la Figura 2.

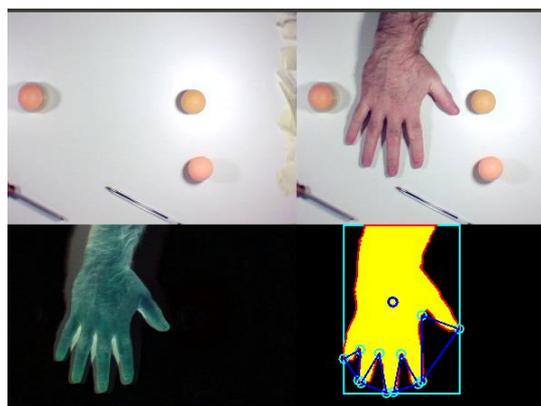


Figura2. Visión Artificial.

2. Servomotor TowerPro SG-5010

TowerPro SG5010 es un servo motor de alta calidad y de muy bajo costo, útil para todas sus necesidades mecatrónicas. Tiene un arnés de alimentación de 3 pines y un cable de control, así como una variedad de componentes extras como se muestra en la Figura 3.



Figura 3. Servomotor TowerPro SG-5010

Características:

Voltaje de funcionamiento: 4.8V ~ 6.0V
 Velocidad de operación:
 0.20sec/60 grados (4.8V)
 0.16sec/60 grados (6.0V)
 Torque de accionamiento:
 5,2 kg * cm (4,8 V)
 6,5 kg * cm (6.0V)
 Rango de temperatura: -20 ° C ~ 60 ° C
 Ancho de banda muerta: 4ms
 5,2 kg * cm (4,8 V)
 6,5 kg * cm (6.0V)
 Dimensión:
 41 x 20 x 38 mm
 Peso: 41 g
 Longitud del cable conector: 30cm

$$\% \text{ Duty Cycle} = \frac{\text{On Time}}{(\text{On Time} + \text{Off Time})}$$

Figura4. Formula de posición de los servos con PWM

3. LABVIEW

Labview es un entorno de programación destinado al desarrollo de aplicaciones, similar a los sistemas de desarrollo comerciales que utilizan, cuyo icono se muestra en la Figura 5. Sin embargo, Labview se diferencia de dichos programas se basan en líneas de texto para crear el código fuente del programa, mientras que Labview emplea la programación grafica o lenguaje G para crear programas basados en diagramas de bloques [3].

Este programa tiene su mayor aplicación en sistemas de medición, como monitoreo de procesos y aplicaciones de control, un ejemplo de esto pueden ser sistemas de monitoreo en transportación, laboratorios para clases en universidades, procesos de control industrial. Labview es muy utilizado en procesamiento digital de señales procesamiento en tiempo real de aplicaciones biomédicas, manipulación de imágenes y audio,

automatización, diseño de filtros digitales, generación de señales, entre otras cosas.

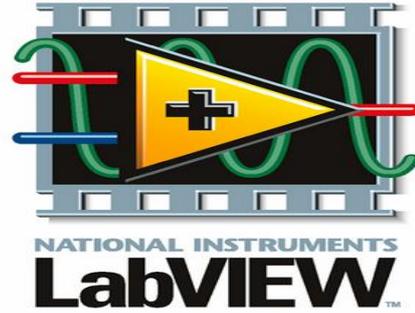


Figura 5. Software Labview

4. Microcontrolador PIC16F877A

4.1. Características del PIC 16F877A [4] :

- Memoria de datos: 368 bytes RAM, 256 bytes EEPROM.
- Dos módulos de Captura, Comparación y PWM: - Captura: 16 bits. Resolución máx. = 12.5 nseg. - Comparación: 16 bits. Resolución máx. = 200 nseg. - PWM: Resolución máx. = 10 bits.
- Convertidor Análogo/Digital de 10 bits multicanal (8 canales de entrada).
- Puerto serial síncrono (SSP) con bus SPI (modo maestro) y bus I²C (maestro/esclavo).
 - Corriente máxima absorbida/suministrada (sink/source) por línea (pin): 25 mA
- Voltaje de alimentación: 3.0 a 5.5 V DC
- Frecuencia oscilador: 20 MHz (máxima)

II. DESARROLLO

A. Diseño Mecánico

Para esta parte primero hacemos los cálculos teóricos utilizando el algoritmo de Denavit – Hartenberg para hallar la cinemática directa de nuestra mano robótica hacemos su diagrama de hilos del mano antropomórfico, tal como se muestra en la Figura 6.

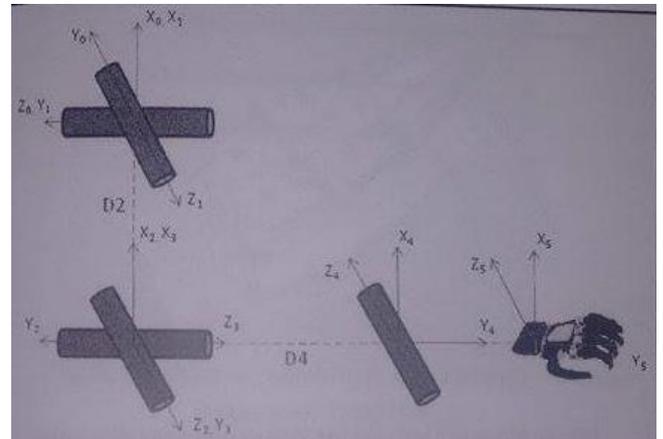


Figura6. Diagrama de hilos del mano antropomórfico

Luego llenamos la tabla de parámetro en la cual se pondrán todos los parámetros de la mano.

Con los valores obtenidos de la tabla reemplazamos los valores en una matriz de Denavit – Hartenberg diferente para cada una de las articulaciones.

i	Θ_i	d_i	a_i	α_i
1	Θ_1	0	0	$\pi/2$
2	Θ_2	0	D2	0
3	Θ_3	0	0	$\pi/2$
4	Θ_4	D4	0	$\pi/2$

$$A_{i-1} = \begin{bmatrix} \cos(\Theta_i) & -\sin(\Theta_i)\cos(\alpha_i) & \sin(\alpha_i)\sin(\alpha_i) & \cos(\Theta_i)\alpha_i \\ \sin(\Theta_i) & \cos(\Theta_i)\cos(\alpha_i) & -\cos(\Theta_i)\sin(\alpha_i) & \sin(\Theta_i)\alpha_i \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Este diseño esta echo por un sistema de mecanizado dando la forma de dedos articulados con el cual podemos darle movimiento mediante unos hilos que están dentro de cada tubo.

Para el cual tenga el correcto movimiento de una mano real se le hace cortes a los tubos en las partes donde estaría las articulaciones interfalangiana, y metacarpofalangiana de la mano para que pueda doblarse de una manera correcta [5].

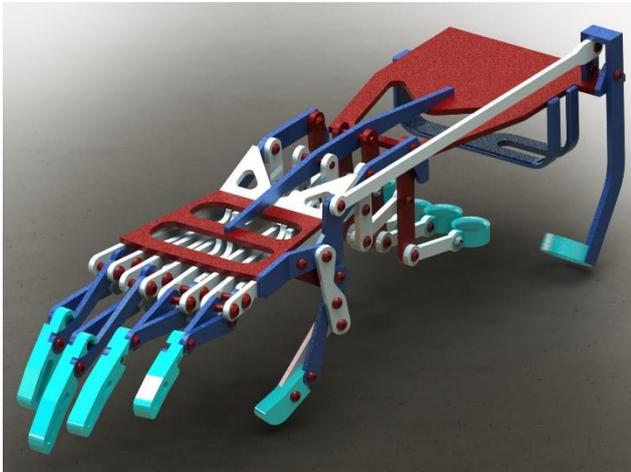


Figura 7. Mano Robótica- Solidworks

Este sistema tendrá la función de darle el movimiento adecuado, dando el movimiento de abrir o cerrar cada dedo.

La mano robótica será de tamaño real de una mano real, estará cubierta con un guante para que se visualice correctamente la mano echa por estos tubos, lo que se muestra en la Figura 7.

B. Diseño Electrónico

En la parte electrónica se usará un sistema de receptor de los datos enviados por el software LabVIEW que será en modulo Bluetooth HC-05 el cual recibirá y mandara al microcontrolador PIC 16F877A la señal adecuada para que el sistema pueda controlar de forma necesaria los 5 servomotores TowerPro SG5010 que estarán

recibiendo señales constantemente para que puedan dar el movimiento que deseamos tener en la mano robótica.

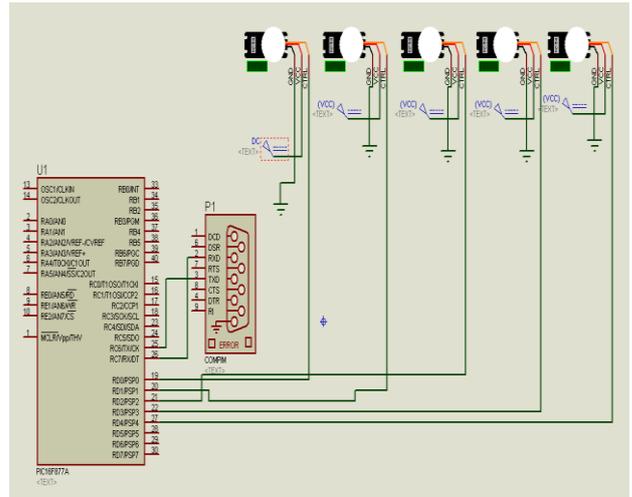


Figura 8. Imagen de la parte electrónica simulada

El programa para utilizar la simulación de los circuitos electrónico con el software fue en PROTEUS y la conexión entre los programas fue el VSPE (Virtual Serial Port Emulator), tal como se muestra en las Figuras 8 y 9.

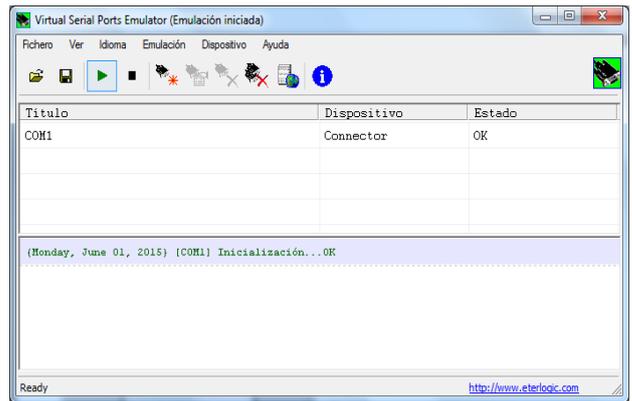


Figura9. Virtual Serial Port Emulator

Este programa hace la simulación de un puerto serial virtual para conectar programa poder simularlos y tener una idea de cómo funcionaría realmente.

C. Programación

En esta parte la programación está basada en 2 partes:

Programación del microcontrolador PIC:

En esta parte usamos el programa CCS (Pic C Compiler) Este programa está basado en lenguaje C para microcontroladores. Este lenguaje es simple y didáctico de utilizar [9].

En nuestro programa se base en sistema de casos para poder optar la mejor señal al PIC.

El cual sea uno de los casos correctos el microcontrolador hace una cierta función para el posicionamiento de los dedos para que nos dé, el resultado esperado.

En este programa utilizaremos la comunicación RS232 del microcontrolador para que pueda conectarse con el ordenador, tal como se muestra en la Figura 10.

```

1 #INCLUDE <16F877A.H>
2 #FUSES XT
3 #USE DELAY (CLOCK=4M)
4 #USE RS232 (BAUD=9600,XMIT=PIN_C6,RCV=PIN_C7)
5 CHAR DATA ; //CHAR TIPO CARACTERDE 8
6 VOID MAIN ()
7 {
8     SET_TRIS_D(0);
9     OUTPUT_D(0);
10    WHILE (TRUE)
11    {
12        DATA= getc();
13        SWITCH (DATA)
14        {
15            CASE 'A':
16                output_d(0b00011111);
17                delay_us(2500);
18                output_d(0b00000000);
19
20            CASE 'B':
21                output_d(0b00011110);
22                delay_us(2500);
23                output_d(0b00000001);
24                delay_us(800);
25                output_d(0b00000000);
26                BREAK;
27
28            CASE 'C':
29                output_d(0b00011111);
30                delay_us(800);
31                output_d(0b00000000);

```

Figura10. Parte del programa en CCS.

Programación del LabVIEW:

Este software es un sistema muy completo y tiene muchas funciones para utilizar como el sistema de reconocimiento de imagen en el cual mediante fotos guardadas en la computadora hacemos comparación con los datos obtenidos en tiempo real con la cámara [3].



Figura11. Imagen Visualizada por el usuario

Esta es la interacción que tiene el programa con el usuario para poder obtener los datos de la mano en el tiempo real, tal como se muestra en la Figura 11.

En esta parte conectamos manualmente la computadora con el módulo BLUETOOTH del circuito para poder mandar la señal al micro controlador.

Este software capta las posiciones de mi mano mediante la cámara para poder mandarle inalámbricamente al PIC.

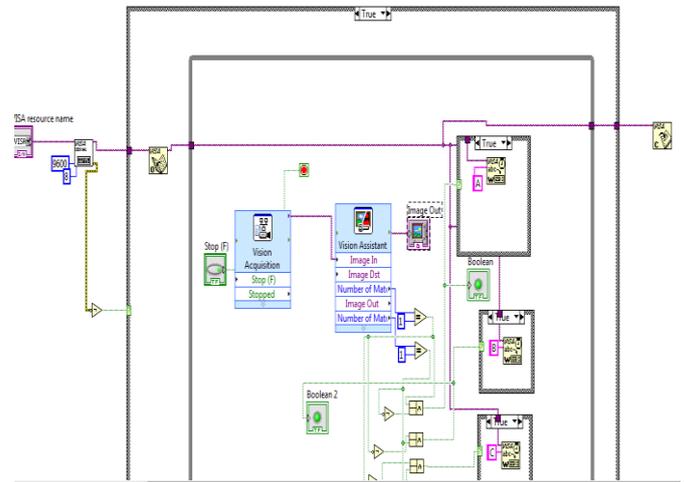


Figura12. Pantalla del Programador echo en LabVIEW

La programación que se utiliza en LabVIEW es hecha por bloques este sistema hace que juntemos los bloques adecuadamente para poder formar una programación sólida y correcta para que tenga ningún error, lo que se muestra en la Figura 12 y 13.

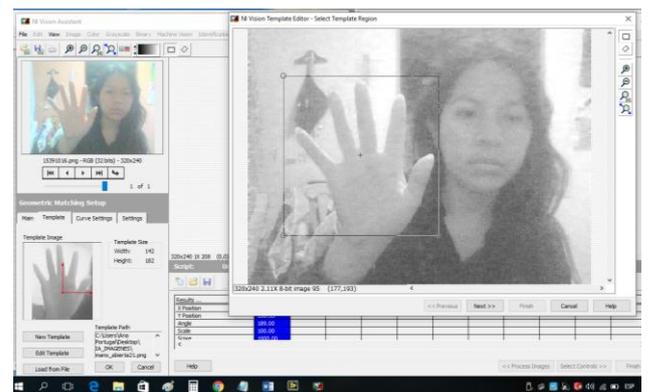


Figura13. Imagen de visualización de la programación mediante el VISION ASSISTANT del LabVIEW

Esta es la parte más importante para el reconocimiento de las formas ya que aquí es donde guardamos las posiciones y las mandamos al microcontrolador en forma de señales para que pueda dar una respuesta adecuada.

III. RESULTADOS

Los resultados obtenidos en este proyecto fueron que la respuesta obtenida por el microcontrolador tiene un pequeño retardo que son milisegundos aproximadamente ya que en el proceso de mandar la información de la computadora al microcontrolador y además de procesar la imagen obtenida hace que se demore un poco.

La posición de la mano puede salir errónea si forma es similar a otra que tenemos guardada en el software, tal como se muestra en la Figura 14.



Figura14. Posiciones similares

Entre más muestra de manos pongamos, el programa cada vez se hace más lento debido a la información que guarda la computadora eso hace que el sistema deje de reconocer en tiempo real y le dé un pequeño retardo para ver todas las muestras que tiene y pueda escoger la más parecida.

IV. CONCLUSIONES

El objetivo de este proyecto se logró conseguir, ya que no solo llego imitarlo si no también llego a realizar la acción de sujetar objetos sólidos.

La función de reconocimiento de visión que está en Labview es eficiente ya que pudo hacer el reconocimiento de la mano.

V. OBSERVACIONES

Tanto la distancia como las condiciones de luz son los factores que pueden llegar a influir en el reconocimiento de la mano.

El desempeño adecuado de la visión artificial corresponde a la calidad y resolución de la cámara.

Este proyecto se realizó a manera de investigación teórica y en base a un prototipo, con la finalidad de una futura implementación como parte de mi proyecto de tesis para optar el grado de Ingeniero Mecatrónico.

VI. RECOMENDACIONES

Tener en cuenta que mientras se opte por dar más muestras a cada tipo de posición de mano.

Para que así se pueda reconocer satisfactoriamente las posiciones de la misma y pueda mandar una señal adecuada al microcontrolador para que la salida sea la más exacta posible.

Esta forma tendrá el fin de que si detecta cualquiera de estas posiciones (que son lo mismo solo que cambian ligeramente por el tamaño y posición) mandara la señal como si fuera una sola y eso lo mandara al microcontrolador para que pueda hacer la posición que mandemos [8].

Entra más muestras tengamos de cada posición de la mano, más preciso será la captación de esta, tal como se muestra en la Figura 15.

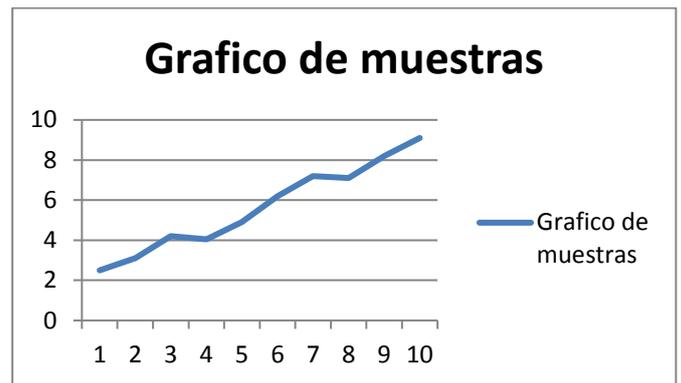


Figura15. Grafico Estadístico de las muestras

Se recomienda también tener la detección de la mano en una caja que puedas controlar la intensidad de la luz ya que esto es un factor muy importante cuando hablamos sobre este tema de reconocimiento de imagen ya que le afecta y puede que no reconozca adecuadamente y nos mande una señal totalmente incorrecta.

VII. RECONOCIMIENTOS

Los autores agradecen los aportes brindados por la universidad Ricardo Palma del Laboratorio de Mecatrónica y docentes en general.

Agradecemos también al apoyo económico del Centro de Investigación de la Universidad Ricardo Palma.

VIII. REFERENCIAS

- [1] Ministerio de salud, “Manual de salud ocupacional”, Lima - Perú, 2005
- [2] Visión Artificial, “Capitulo 1: Introducción a la visión artificial: La naturaleza de la luz visión humana”, <http://www.etitudela.com/celula/downloads/visionartificial.pdf> Junio 2010.
- [3] National Instrumets , “Software de Desarrollo de Sistema NI LabVIEW”, National Instrumet Corporation, 2015.
- [4] Microchip Data Sheet PIC16F87XA 28/40/44-Pin Enhanced Flash Microcontrollers.
- [5] Comité Español de Automática, “Libro Blancode la robótica” CEA-GTRob,2008.
- [6] Servicio de Información y Noticias Científicas (2012). “Las sustancias químicas peligrosas causan más muertes que los accidentes de trabajo” – España
- [7] Mano robótica controlada con el pensamiento, Japón <http://www.novaciencia.com/2006/05/31/disenan-mano-robot-manejada-con-la-mente/>

[8]http://robotsgentina.com.ar/Actuadores_manos.html

[9] Dogan Ibrahim, Advanced PIC Microcontrollers Projects in C from USB to Zigbee with the PIC 18F Series, Elsevier Science & Technology Rights Department in Oxford, 2008, p. 31.

[10] Daniel Alejandro Zúñiga Tenesaca y Diego Miguel Andrade Zeas (2011). Diseño y construcción de una mano robótica para la enseñanza del alfabeto dactilológico universal para personas sordomudas. Artículo Científico de Ingeniería Electrónica, Escuela de Ingeniería, Universidad Politécnica Salesiana, Ecuador.