

# Diseño e implementación de un sistema traductor de lenguaje de señas de manos a un lenguaje de texto mediante visión artificial en un ambiente controlado

Edwin Chiguano, Nathaly Moreno, Luis Corrales

Facultad de Ingeniería Eléctrica y Electrónica, EPN  
Quito, Ecuador

edwinchiguano3@hotmail.com  
natymd9@hotmail.com  
luis.corrales@epn.edu.ec

**Abstract**— En este proyecto se ha realizado un sistema que traduce de lenguaje de señas a lenguaje de texto usando visión artificial, para permitir que personas con discapacidad auditiva y/o visual que manejan el lenguaje de señas puedan comunicarse con el resto de personas que no manejan este lenguaje. Para personas que no conocen este lenguaje el proyecto ofrece la opción de un entrenador, para que aprendan cada uno de los símbolos de este lenguaje y adquieran la habilidad necesaria para utilizar la aplicación.

El sistema adquiere la imagen que luego pasa por un procesamiento digital de imágenes y por último se realiza la traducción. El programa que fue desarrollado en Labview 2009 incluyendo su toolkit de visión artificial.

En el procesamiento digital de imágenes se aplicaron algunos filtros y operaciones morfológicas para resaltar las características de la imagen y eliminar información innecesaria como ruido. También se eliminaron objetos extraños en la imagen mediante un recortado del área de interés.

Con la ayuda del Vision Assistant de Labview se elaboraron las bases de datos, para llevar a cabo la comparación con la imagen recortada y de esta manera asignar la clase (letra) correspondiente a cada imagen.

Con la clase asignada se forma el texto que se muestra en forma escrita en la pantalla o a su vez se puede enviar a un documento de Word, además es posible reproducir en audio el texto formado con la ayuda de la herramienta “texto a voz” de Windows.

*Índices – Traductor de señales de manos a texto mediante visión artificial.*

## 1 INTRODUCCIÓN

El objetivo del presente proyecto es desarrollar un sistema que reconozca las señales de manos usadas por personas con discapacidad y traducirlas a texto y audio. El

reconocimiento de las señas se realiza mediante técnicas de visión artificial, usando una cámara web y el software Labview.

### A. Lenguaje de señas

El lenguaje de señas, lenguaje dactilológico o lengua de signos es una forma de expresar caracteres, palabras o frases mediante signos usando principalmente las manos con el fin

de comunicarse sin la necesidad de un sistema de signos vocales.

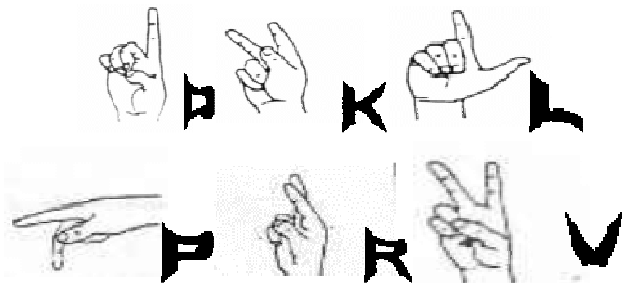


Fig. 1. Alfabeto de Señas Ecuatoriano [1].

Existen varios tipos de lenguaje de signos, varían dependiendo del país, estado, provincia e incluso ciudad, por lo que no existe un lenguaje de señas universal. Países como: Venezuela, México, España, Estados Unidos, Argentina, Perú, Ecuador y otros tienen su propio lenguaje de señas. Ecuador menciona la lengua de señas ecuatoriana en el artículo 53 de la Constitución Política de 1998 [2].

### B. Visión artificial

La visión artificial se desarrolla como una rama del estudio de la inteligencia artificial. La visión artificial tiene como propósito programar un computador para que pueda interpretar la información que contiene una imagen.

Un sistema de visión artificial consta de los siguientes elementos:

**Iuminación:** Consta de todos los dispositivos (lámparas, láser, LEDs, etc.) que sirven como fuente de luz, y a través de los cuales se va a tener una iluminación lo más uniforme posible. De este subsistema depende en gran parte la adecuada adquisición de la imagen.

**Captación:** En este subsistema se encuentran los transductores que convierten la radiación luminosa reflejada en señales eléctricas, por ejemplo las cámaras CCD.

**Sistema de Adquisición:** En este subsistema se transforma las señales eléctricas que provienen de las cámaras en señales de video, esto se realiza por medio de tarjetas de adquisición colocadas en el computador.

**Procesamiento:** Consta de uno o varios computadores en los cuales se analiza la imagen captada por el subsistema de adquisición. Con la ayuda de algunos algoritmos es posible obtener la información más relevante, la cual nos permite posteriormente adecuarla a una aplicación específica.

**Periféricos:** Aquí se ubican todos los equipos o dispositivos que reciben la información del computador y cumplen la función de actuadores dentro del proceso o aplicación.

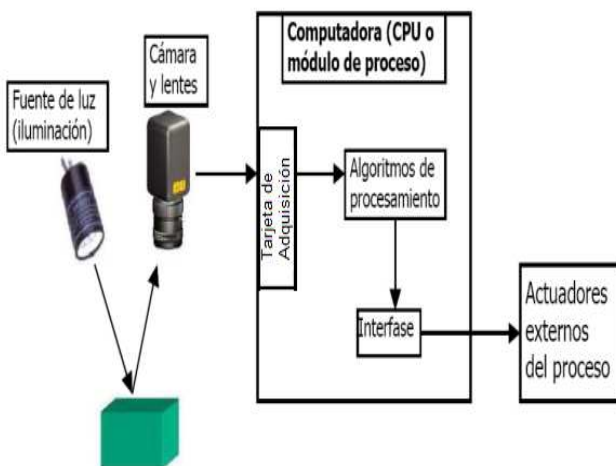


Fig. 2. Elementos de un sistema de visión artificial (Tomado de [3]).

### C. Software para visión artificial

Existen una gran cantidad de software que permiten realizar aplicaciones de visión artificial como por ejemplo: Matlab, Labview, Opencv, VisualBasic, etc. Para la realización de este proyecto se utilizó el software de Labview con los toolkits de visión artificial, porque permite realizar aplicaciones de forma simple gracias a su programación gráfica amigable con el usuario. Además se adapta fácilmente al hardware (cámaras USB), que se está

utilizando y posee una aplicación dedicada al reconocimiento de imágenes, que es el objetivo de este proyecto.

### D. Descripción del traductor de señales de manos mediante visión artificial

El sistema de traducción está dividido en varios procesos, estos procesos se concatenan consecutivamente para lograr cumplir con el objetivo del proyecto.

Para reducir el tiempo de ejecución del programa se redujo el procesamiento de la imagen, para esto la toma de la imagen se la realizo en un ambiente donde el objeto de interés (mano) contraste con el resto de la imagen. Por este motivo se escogió el ambiente de la Figura 3.



Fig. 3. Fondo oscuro para contraste de la mano.

El sistema de iluminación que se escogio es difusa para que este no deslumbre a la cámara, la luminaria esta compuesta de un reflector y una lampara fluorescente que se alimenta a 110 Vac como se ve en la Figura 4.



Fig. 4. Sistema de iluminación difuso.

Este sistema de iluminación es complemento al sistema de iluminación existente en el lugar donde se usa la aplicación.

El sistema de contraste nos sirve para mejorar la calidad de información que contiene la imagen, de igual forma se reduce el tiempo de la aplicación, dado que la imagen no necesita un algoritmo de segmentación complejo.

El software del traductor consta de las siguientes etapas: detección del objeto de interés, paso a escala de grises y mejoramiento de contraste, proceso de binarización, clasificación y ensamblaje de la traducción.

### Detección del objeto de interés

Esta etapa tiene como finalidad ejecutar la aplicación siempre y cuando exista un objeto sobre el área que está captando la cámara. Este análisis se lo realiza utilizando el histograma de la imagen, se cálculo la media de los valores de los pixeles, estos valores se comparan con los del área de trabajo y se supervisa constantemente hasta que se produce el cambio en estos valores, lo que indica que existe objeto sobre el área de trabajo, dando paso a la siguiente etapa de la aplicación. La diferencia en el valor de los pixeles se puede ver en la Figura 5 y en la Figura 6.

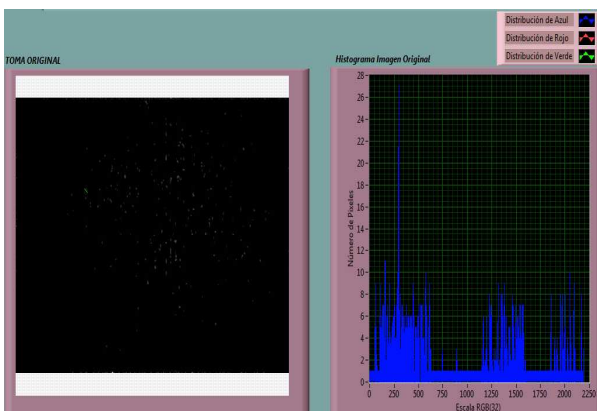


Fig. 5. Imagen e histograma del fondo (Sin mano).

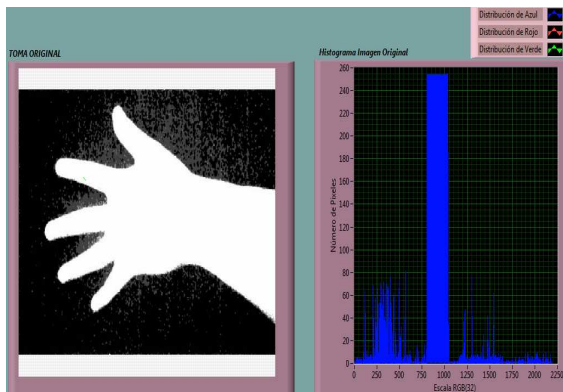


Fig. 6. Imagen e histograma con presencia de mano.

### Paso a escala de grises y mejoramiento de contraste

Una vez detectado el objeto de interés, se pasa la imagen a escala de grises porque la aplicación no requiere de un análisis del color, pero sí de su forma. Con esto se tiene una imagen de menor tamaño para su posterior proceso de segmentación. Se realiza un contraste a la imagen con la finalidad de eliminar el ruido existente y resaltar la forma del objeto dentro de la imagen como se observa en la Figura 7.

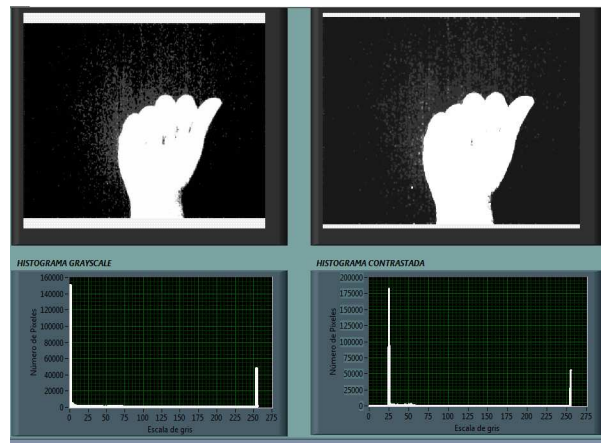


Fig. 7. Imagen en escala de grises y mejoramiento del contraste.

### Proceso de binarización

Este proceso tiene como objetivo convertir la imagen en escala de grises a una imagen binaria, donde los pixeles tengan dos valores, ya sea 1L o 0L. Este consiste en escoger un umbral basado en los valores de los pixeles que se observa en el histograma, el umbral se debe escoger para que el objeto de interés este dentro de la imagen. Una vez escogido este umbral la imagen resultante es una donde solo existen dos valores de color Blanco y Negro. Para eliminar el ruido existente en la imagen binaria se realiza operaciones morfológicas como son erosión (1), dilatación (2), filtros en el dominio de la frecuencia y filtros basados en morfología de los objetos con los cuales se elimina objetos innecesarios dentro de la imagen.

$$A \ominus B = \{x \in E^N | x = x + b \in A \text{ para todo } b \in B \quad (1)$$

$$A \oplus B = \{c \in E^N | c = a + b \text{ para todo } a \in A \text{ y } b \in B \quad (2)$$

El filtro de frecuencia pasa bajos tiene la función de eliminar ruido existente en los bordes de la imagen, funciona como un suavizador de bordes en el dominio espacial, para usar un filtro en el dominio de la frecuencia se sigue los pasos que se indican en la Figura 8.

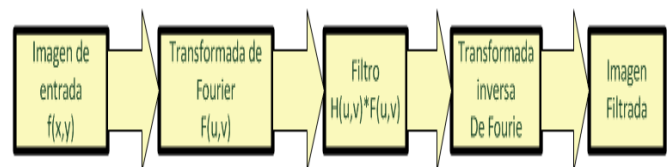


Fig. 8. Etapas del procesamiento de imágenes en el dominio de la frecuencia



Fig. 9. Imagen resultante del proceso de binarización.

En esta etapa también se realiza el recortado del objeto de interés, esto se realiza con la finalidad de optimizar el proceso de clasificación, porque solo se envía el objeto de interés para ser clasificado y no toda la imagen como se ve en la Figura 10.

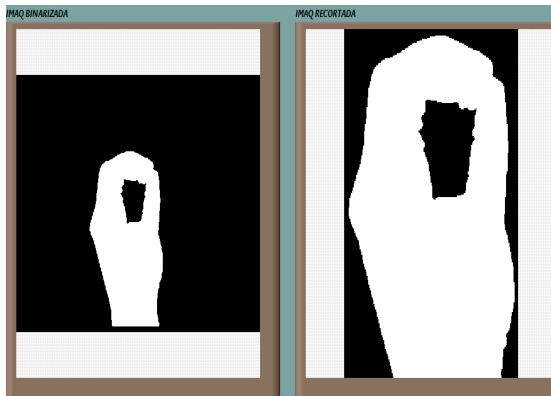


Fig. 10. Imagen binarizada y recortada.

## Clasificación

Esta etapa tiene como finalidad extraer características del objeto de interés y ordenarlas en un vector. Este vector se compara con los vectores existentes en la base de datos para asignar una clase.

En el proceso de clasificación se utilizó el método del K vecino más cercano. Este es un método de clasificación no paramétrico, se basa en la estimación de la densidad de probabilidad o, directamente, en la probabilidad que X pertenezca a determinada clase. Este algoritmo es más robusto frente al ruido que el algoritmo del vecino más cercano.

Consiste en un sistema de votación que depende del número k de muestras que se tomen para realizar la clasificación.

Se toma un número k de las muestras más cercanas, de estas k muestras cercanas se cuentan las de mayor número que pertenezcan a una misma clase y de acuerdo a esto se asigna la clase.

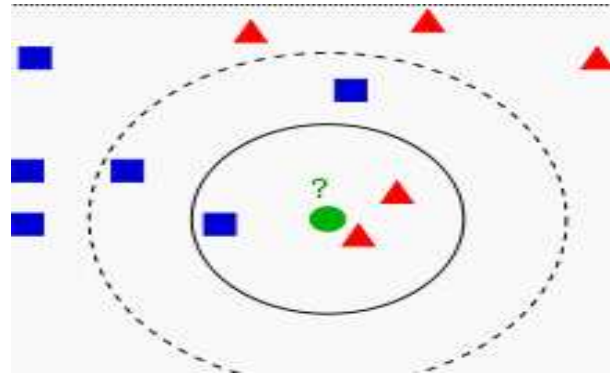


Fig. 11. Ejemplo del Algoritmo K-vecino más cercano [4].

De la Figura 1.25 se observa que para el círculo de menor radio, con  $k = 3$ , el círculo es clasificado con los triángulos ya que estos tienen el mayor número de muestras ( $m=2$ ); en cambio para el círculo de mayor radio con  $k= 5$ , el círculo es clasificado con los cuadrados, porque corresponde al mayor número de muestras ( $m=3$ ).

Para mejorar el proceso de clasificación se realiza el cálculo de la distancia entre vectores característicos, con tres formas distintas, estas formas son:

**Distancia Euclídeana (Euclidean)** es menos sensible a pequeñas variaciones entre muestras, se usa cuando se requiere clasificar muestras con pequeñas variaciones en la misma clase. Su ecuación se ve en (3):

$$d(X, Y) = \sqrt{\sum_{i=1}^n (X_i - Y_i)^2} \quad (3)$$

**Distancia Suma (Sum)** es conocida también como la Métrica Manhattan, es la más usada en procesos de clasificación y viene seleccionada por default en el "Vision Assistant". Se refiere al promedio de distancias, se usa en formas definidas. Su ecuación se ve en (4):

$$d(X, Y) = \sum_{i=1}^n |X_i - Y_i| \quad (4)$$

**Distancia Máxima (Maximum)** es el más sensible a pequeñas variaciones entre muestras, se usa cuando se requiere clasificar muestras con pequeñas variaciones en diferentes clases. Su ecuación se ve en (5):

$$d(X, Y) = \max_i |X_i - Y_i| \quad (5)$$

Con la utilización de estos métodos el proceso de clasificación es robusto y seguro, estos métodos funcionan en paralelo como se indica en la Figura 12.



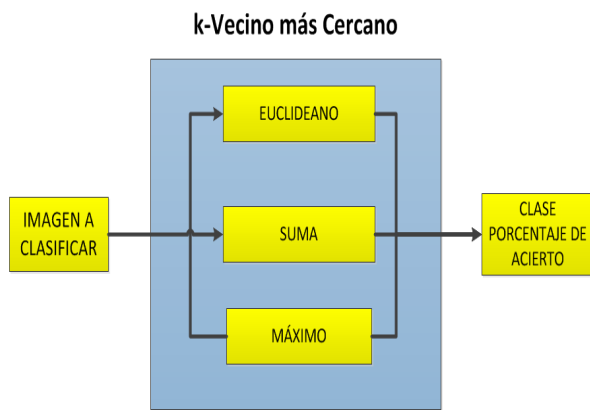


Fig. 12. Diagrama de bloques del proceso de clasificación.

En resumen el proceso de clasificación utiliza un solo método con tres formas distintas para el cálculo de la distancia.

Las bases de datos con cada método para el cálculo de la distancia se genera en el software Vision Assitant.

### Ensamblaje de la traducción

Esta es la última etapa de la aplicación donde se reúnen las clasificaciones aceptadas como correctas y se las une en una frase, interpretando así lo que el usuario desea expresar.

El software tiene la opción de enviar esta frase a un documento de Word y también de reproducirlo en forma de audio. Para reproducir en forma de audio se unió el software del traductor realizado en Labview con la herramienta "Texto a voz" que tiene el Windows mediante nodos de propiedad y el uso del control ActiveX del reproductor.

La pantalla del sistema terminado se observa en la Figura 13.



Fig.13. Imagen clasificada y texto formado.

Utilizando estos subprocesos del traductor se implementó un entrenador, para que personas que no tengan conocimiento del lenguaje de señas lo aprendan y se adiestren en la posición que debe tener la mano para cada letra. El panel frontal para el entrenador es el de la Figura 14.



Fig. 14. Panel frontal del entrenador en modo automático.

Este entrenador puede trabajar en modo automático y manual, en modo automático el software genera aleatoriamente una letra con su respectivo signo para que el usuario la realice y la aprenda y en modo manual el usuario selecciona la letra que desea aprender.

### E. Pruebas del traductor

Con el sistema completo se realizó pruebas de reconocimiento de las señas, los resultados que se obtuvieron se presentan en la Tabla 1.

TABLA I  
ÍNDICES DE ACIERTO PARA CADA LETRA.

K-Vecino más Cercano (índices de acierto)				
Letra	Euclidean	Maximum	Sum	Total
A	1000,00	1000,00	1000,00	749,94
B	1000,00	1000,00	1000,00	749,94
C	1000,00	1000,00	500,00	749,94
D	1000,00	1000,00	1000,00	749,94
E	500,00	500,00	500,00	500,05
F	1000,00	500,00	1000,00	749,94
G	1000,00	1000,00	1000,00	749,94
H	1000,00	1000,00	500,00	749,94
I	1000,00	1000,00	1000,00	749,94
K	1000,00	1000,00	1000,00	749,94
L	1000,00	1000,00	500,00	749,94
M	1000,00	1000,00	1000,00	749,94
N	1000,00	1000,00	500,00	749,94
O	1000,00	1000,00	500,00	749,94
P	1000,00	1000,00	1000,00	749,94
Q	1000,00	1000,00	1000,00	749,94
R	1000,00	500,00	1000,00	749,94
S	500,00	1000,00	500,00	749,94
T	1000,00	1000,00	1000,00	749,94
U	500,00	500,00	1000,00	749,94
V	1000,00	1000,00	1000,00	749,94

<b>K-Vecino más Cercano (índices de acierto)</b>				
<b>Letra</b>	<b>Euclidean</b>	<b>Maximum</b>	<b>Sum</b>	<b>Total</b>
<b>W</b>	500,00	<b>No reconoce</b>	500,00	500,05
<b>X</b>	500,00	500,00	1000,00	749,94
<b>Y</b>	1000,00	1000,00	1000,00	749,94
<b>PROMEDIO</b>	<b>895,83</b>	<b>891,30</b>	<b>833,33</b>	<b>729,12</b>

Con estos resultados se comprueba el correcto funcionamiento de la aplicación.

Se debe tener en cuenta que el rendimiento de la aplicación depende de la habilidad que el usuario posea para realizar los signos correspondientes a cada letra.

La velocidad de ejecución de la aplicación también se puede variar, de esta forma el usuario puede configurar el traductor de acuerdo a su habilidad para realizar los signos de la mano.

#### *F. Conclusiones y Recomendaciones*

##### **Conclusiones**

Un sistema de iluminación correctamente seleccionado, reduce altamente el proceso de análisis de la imagen ya que una imagen es el reflejo de la luz. De esto se obtiene un mejor resultado dentro de los límites de la aplicación reduciendo la complejidad del sistema y su tiempo de ejecución.

El uso de Labview para el desarrollo del software permitió realizar una programación sencilla y ordenada, además que se pudo realizar una interfaz de fácil uso y amigable con el usuario.

Del buen desempeño del algoritmo de preprocesamiento de la imagen depende el proceso de clasificación, porque el preprocesamiento resalta las características de la imagen que el clasificador necesita.

Realizando el proceso de segmentación morfológica de la imagen antes de ingresarla en la base de datos, permite obtener vectores de características definidos en cada clase, lo cual nos evita errores al momento de la clasificación.

Labview es un programa muy completo ya que su toolkit de Visión es una herramienta que contiene todas las operaciones en cuanto al procesamiento de imágenes se refiere, lo que permitió desarrollar totalmente la aplicación en Labview sin necesidad de software adicional.

Del conocimiento y habilidad que el usuario tiene para realizar los símbolos, depende el reconocimiento y por tanto, la asignación de la clase con un porcentaje de error bajo.

Como el usuario requiere facilidad y comodidad a la hora de usar la aplicación, se construyó un hardware que ofrezca facilidades de adaptación a diferentes usuarios, además que es fácilmente transportable y montable.

Se investigaron los métodos de clasificación de Vecino más Cercano, K- Vecino más Cercano y Mínima Distancia; de los cuales se concluye: para aplicaciones donde existe ruido el método el vecino más cercano es poco efectivo, en aplicaciones con mucho ruido y cuyos patrones tiene variaciones muy pequeñas entre si es mejor usar el método de la mínima distancia. Por lo tanto para esta aplicación con presencia de ruido y cuyos patrones en su mayoría están bien diferenciados se usa el método del K-Vecino más Cercano.

##### **Recomendaciones**

Para poder añadir a esta aplicación las letras que poseen movimiento se podría usar un módulo RIO de la National Instrument, el cual permite trabajar el procesamiento de imágenes en tiempo real y tiene la capacidad de usar cámaras más rápidas como las Smart, lo que permitiría usar un algoritmo para el reconocimiento de trayectorias en tiempo real.

El reconocimiento de formas se puede usar en otros procesos en los cuales se necesite reconocer formas definidas. Esto se lograría creando bases de datos con las formas que se desee reconocer, para lograr una automatización en estos procesos.

Para empezar a desarrollar una aplicación similar se recomienda estudiar los métodos de clasificación de imágenes existentes, sus ventajas y desventajas, para seleccionar el método correcto para la aplicación que se desee realizar.

Al momento de escoger el ambiente donde se va a realizar un sistema de visión artificial se debe tener en cuenta: los factores externos e internos que afectan al sistema por ejemplo la velocidad de la aplicación para diseñar correctamente cada uno de los elementos del sistema de visión artificial.

En un trabajo futuro para mejorar el rendimiento del traductor, se debería añadir un algoritmo de predicción de texto para aumentar la velocidad de la aplicación.

Para obtener mejores resultados en esta aplicación, se deben crear las bases de datos dentro del Vision Assistant en base a un preprocesamiento y procesamiento dentro del mismo paquete, porque así se resalta la morfología de la mano, permitiendo crear vectores de características con valores más definidos. Al emplear estas bases de datos en el reconocimiento se logra altos índices de acierto, que a su vez favorece a la asignación de clase con menos porcentaje de error (confusión entre letras).

Al usar el método del K-Vecino más Cercano se recomienda poner un número de K menor al número de muestras de cada clase, de esta forma se logra que el sistema compare las K muestras más parecidas a la imagen que adquiere la cámara. Por lo tanto asigna una clase con alto porcentaje de acierto reduciendo así el error.

Un factor importante al momento de elegir la cámara es si esta tiene o no la función del autocontraste debido a que esta función no permite al sistema funcionar correctamente, porque varía la intensidad de los píxeles. Si la cámara posee esta característica, se la debe eliminar para que este parámetro no afecte al sistema.

Para el sistema de adquisición de imagen se usó una cámara web de 30fps que por sí misma mete un retardo en la adquisición limitando así la adquisición en tiempo real, además que el procesamiento de cada imagen retarda aún más la respuesta del sistema. Para mejorar esta latencia en la aplicación se debería usar cámaras especializadas en visión artificial que adquieren imágenes a mayor velocidad.

Áreas de interés: robótica móvil, desarrollo de software, microcontroladores, automatización y control industrial. (edwinchiguano3@hotmail.com)

## G. REFERENCIAS

- [1] Sitio de Sordos, (2011) [Online], Disponible: [http://www.sitiodesordos.com.ar/alfabeto\\_ecua.htm](http://www.sitiodesordos.com.ar/alfabeto_ecua.htm)
- [2] Wikipedia, (2010) [Online], Disponible: <http://es.wikipedia.org/wiki/Dactilolog%C3%ADa>
- [3] U. Salesiana, (2007) Dspace, [Online], Disponible: <http://dspace.ups.edu.ec/bitstream/123456789/220/2/Capitulo%201.pdf>
- [4] Wikipedia, (2011) [Online], Disponible: <http://es.wikipedia.org/wiki/Knn>

## H. BIOGRAFIAS



**Nathaly Moreno**, nació en Quito-Ecuador el 2 de Julio de 1986. Realizó sus estudios secundarios en el Colegio Femenino "Espejo", donde obtuvo el título de Bachiller especialidad Físico Matemático. Se graduó en la Escuela Politécnica Nacional como Ingeniero en Electrónica y Control en 2011. Actualmente desempeña el cargo de Ayudante del

Laboratorio de Sistemas de Control, perteneciente al Departamento de Automatización y Control Industrial (DACI) de la Escuela Politécnica Nacional.

Áreas de interés: control de procesos, neumática y automatización y control industrial. (natymd9@hotmail.com)



**Edwin Chiguano**, nació en Quito-Ecuador el 16 de Febrero de 1987. Obtuvo su título de bachiller técnico industrial en electricidad en el Instituto Tecnológico Superior Sucre. Se graduó en la Escuela Politécnica Nacional como Ingeniero en Electrónica y Control en 2011. Desempeña el cargo de asistente técnico administrativo en el Observatorio Astronómico de

Quito desde 2009 hasta la actualidad.