

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
INGENIEROS DE SISTEMAS**

**TEMA:
ESTUDIO DE PROCEDIMIENTOS Y ALGORITMOS PARA EL
RECONOCIMIENTO AUTOMÁTICO DE LAS CARTAS DE UN JUGADOR DE
40 NO VIDENTE**

**AUTORES:
RENÉ FERNANDO GUALLI USHIÑA
OSCAR SANTIAGO ANGAMARCA PUPIALES**

**DOCENTE TUTOR:
RODRIGO EFRAÍN TUFÍÑO CÁRDENAS**

Quito, Septiembre 2016

CESIÓN DE DERECHOS DE AUTOR

Nosotros René Fernando Gualli Ushiña con documento de identificación N° 1718301540 y Oscar Santiago Angamarca Pupiales, con documentos de identificación N° 1002659678, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: “ESTUDIO DE PROCEDIMIENTOS Y ALGORITMOS PARA EL RECONOCIMIENTO AUTOMÁTICO DE LAS CARTAS DE UN JUGADOR DE 40 NO VIDENTE”, mismo que ha sido desarrollado para optar por el título de: INGENIERO DE SISTEMAS, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

.....
RENÉ FERNANDO
GUALLI USHIÑA
CI: 1718301540

.....
OSCAR SANTIAGO
ANGAMARCA PUPIALES
CI: 1002659678

Fecha: Septiembre de 2016

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el artículo académico, ESTUDIO DE PROCEDIMIENTOS Y ALGORITMOS PARA EL RECONOCIMIENTO AUTOMÁTICO DE LAS CARTAS DE UN JUGADOR DE 40 NO VIDENTE realizado por René Fernando Gualli Ushiña y Oscar Santiago Angamarca Pupiales, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerados como trabajo final de titulación.

Quito, Septiembre de 2016



RODRIGO EFRAÍN TUFIÑO CÁRDENAS

CI: 1717646390

ÍNDICE

I INTRODUCCIÓN	1
II. MATERIALES Y MÉTODOS	2
A. Modulo 1: Tratamiento y segmentacion de imagen.....	2
1) Adquisición de imágenes	2
2) Escala de Grises	2
3) Filtro.....	3
4) Binarización	3
5) Segmentación.....	3
B. Módulo 2: Reconocimiento de caracteres mediante una Red Neuronal Artificial.....	4
1) Datos de entrada a la RNA.....	4
2) Implementación de la Red Neuronal Artificial	4
3) Entrenamiento de una RNA para el reconocimiento de caracteres.....	5
4) Reconocimiento de patrones	6
III. RESULTADOS Y DISCUSIÓN	6
A. Determinación de capas ocultas en la RNA.....	6
B. Precisión del reconocimiento	6
IV CONCLUSIONES Y TRABAJO FUTURO.....	8
REFERENCIAS	8
ANEXOS	10
Anexo 1. Características técnicas cámara.....	10
Anexo 2. Características técnicas mini CPU	11
Anexo 3. Fotografías del Prototipo.....	12
Anexo 4. Interfaz gráfica	14

ÍNDICE DE FIGURAS

Figura 1. Estructura el sistema de reconocimiento de cartas.....	2
Figura 2. Etapas a seguir para el tratamiento de la imagen	2
Figura 3. Dispositivo de reconocimiento automático de cartas	2
Figura 4. Imagen real capturada por la cámara.	2
Figura 5. Imagen en escala de grises	3
Figura 6. Imagen aplicada filtro Gausiano.	3
Figura 7. Imagen binarizada	3
Figura 8. Segmentación por detección de contornos	3
Figura 9. Relleno de contornos	3
Figura 10. Ejemplo de segmentación para la carta “J”	4
Figura 11. Área de interés segmentada e invertida.....	4
Figura 12. Módulo 2, Reconocimiento de caracteres usando una RNA.....	4
Figura 13. Caracter “J” representado en una matriz 16x16.....	4
Figura 14. Valor de cada pixel, para obtener al vector.....	4
Figura 15. Esquema de un MLP	5
Figura 16. Tipos de representación del caracter “J”	5
Figura 17. Representación de datos de entrada y salida para el reconocimiento del caracter “J”	6
Figura 18. Número de neuronas capa oculta vs. Efectividad [%]	6
Figura 19. Número de neuronas capa oculta vs. Tiempo de entrenamiento [seg]	6
Figura 20. Histograma de efectividad para cada carta.....	8

ÍNDICE DE TABLAS

Tabla I. Valores de contornos para la carta “J”	4
Tabla II. Matriz de confusión sin luz artificial	7
Tabla III. Matriz de confusión con luz artificial	7

RESUMEN

El presente documento realiza una investigación enfocada a las personas con discapacidad visual que han sido víctimas de discriminación durante años, renunciando a sus derechos dentro de la comunidad. Este artículo pretende promover la inclusión de personas no videntes al entretenimiento social, como lo son los juegos de cartas y específicamente el juego popular en Ecuador llamado “cuarenta “a través de notificaciones auditivas provenientes de un ordenador. Basado en nuevos procedimientos y algoritmos robustos de visión artificial para obtener los atributos necesarios de una imagen digitalizada; y el entrenamiento de redes neuronales artificiales multicapa para el reconocimiento automático de cada carta: es posible el reconocimiento automático de cartas en un ambiente diseñado predeterminado con una fiabilidad superior al 85 % y aumentando su rendimiento al 94.7% al incluir luz artificial. Todo esto utilizando herramientas de software libre como Open CV, Qt Creator y C++, bajo una plataforma de implementación Arch Linux.

ABSTRACT

This paper makes a research focused on people with visual disabilities have been discriminated over the years and they have resigned their rights into the community. This paper pretends to promote social inclusion to people with visual impairments by means of card games. Specifically a popular game in Ecuador called "Cuarenta" which will be assisted by auditive notifications coming from a computer. In order to get the required attributes to obtain a digitalized image, this project is based on new procedures and robust algorithms of artificial vision, and artificial multi-layer neural network training for automatic card recognition. Applying this techniques; at a preset designed environment, it is possible to automatically recognize a card with a reliability greater than 85% and increasing its reliability to 94.7% if artificial light is included. All of these features can be achieved using open source software as Open CV, Qt Creator and C++, under an implementation platform Arch Linux.

Study of Procedures and Algorithms for the Automatic Recognition of the Cards of a Blind Player of 40

R. Gualli¹; O. Angamarca¹; R. Tufiño¹; H. Ortega¹ and A. Zambrano²

Abstract— People with visual disabilities have been discriminated over the years and they have resigned their rights into the community. This paper pretends to promote social inclusion to people with visual impairments by means of card games. Specifically a popular game in Ecuador called "Cuarenta" which will be assisted by auditive notifications coming from a computer. In order to get the required attributes to obtain a digitalized image, this project is based on new procedures and robust algorithms of artificial vision, and artificial multi-layer neural network training for automatic card recognition. Applying this techniques; at a preset designed environment, it is possible to automatically recognize a card with a reliability greater than 85,10% and increasing its reliability to 94,70% if artificial light is included. All of these features can be achieved using open source software as Open CV, Qt Creator and C++, under an implementation platform Arch Linux.

Keywords— Artificial vision, Artificial neural networks, Backpropagation, OpenCv.

I INTRODUCCIÓN

LA VISIÓN Artificial (VA) o visión por computadora [1] [2], es un campo científico que se desarrolla dentro de la Inteligencia Artificial (IA) [3], la cual permite analizar en detalle la información que puede ser obtenida a partir de un conjunto de vistas en una escena, permitiendo así, establecer la relación entre el mundo tridimensional y sus vistas bidimensionales tomadas por él. Su metodología, incorpora la adquisición, procesamiento, análisis e interpretación de imágenes reales con la finalidad de reconstruir el espacio tridimensional a partir de sus vistas. La VA es una herramienta en auge y ampliamente utilizada, tanto que grandes compañías como *Google*, *Facebook* y *Twitter* han invertido grandemente en este tema a través de investigaciones como: algoritmos de aprendizaje automático, algoritmos genéticos, algoritmos de entrenamiento, etc.; con el fin de mejorar su productividad, la

calidad de su servicio y por ende, la experiencia del usuario. El usuario en todo sistema informático es un ente esencial y prioritario, y este artículo se encuentra enfocado a un sector de usuarios olvidados por las Tecnologías de la Información, las personas no videntes. Solamente en el Ecuador, según cifras del Instituto Nacional de Estadísticas y Censos (INEC) [4], existen alrededor de 274.000 personas no videntes en el país, las cuales presentan dificultades para mantener un ritmo de vida normal cotidiano; y mucho más, el pelear por la integración dentro de una sociedad vidente, como lo es el compartir una lectura, las actividades deportivas, o los juegos de mesa; y es en este último, un juego de mesa llamado 40, que este artículo enfoca su investigación y desarrollo.

El juego de "Cuarenta" es un juego de cartas de mesa popular en el Ecuador. Dos o cuatro jugadores intervienen en el proceso jugando en 2 equipos situados en extremos; todos formando una cruz. Cada persona comienza con 5 cartas y el objetivo es "ganar puntos" hasta llegar a 40 puntos al lanzar las cartas adecuadas siguiendo determinadas reglas como se detalla en [5]. Es importante citar que el color de los caracteres y las cartas 8, 9 y 10 no son parte del juego.

La persona no vidente comienza su juego con 5 cartas que ingresa a un prototipo adaptado para facilitar su manipulación. Este prototipo dispone de cinco ranuras donde cada carta es introducida, dejando únicamente al descubierto la sección donde se ubica el caracter que será reconocido por el sistema. El objetivo es identificar por medio de una cámara web los caracteres de las cartas físicas de izquierda a derecha de una imagen digital, para un jugador realizando el tratamiento de la imagen. El proceso utiliza técnicas de procesamiento digital de imágenes como binarización y segmentación [6] [7], para reconocer de forma automática el caracter de cada carta, a través de una Red Neuronal Artificial (RNA) [8] [9] [10] previamente entrenada, y obtener como resultado la información del caracter en un formato de voz que el usuario no vidente logre captar.

Por tanto, el diseño y desarrollo del sistema involucra el incursionar en nuevos desafíos dentro de la IA haciendo uso de herramientas de *software* libre, como: librerías de *OpenCV* [11], plataforma de implementación *Qt Creator* y lenguaje de

¹ R. Gualli, Universidad Politécnica Salesiana, Quito, Ecuador, rgualli@est.ups.edu.ec

O. Angamarca, Universidad Politécnica Salesiana, Quito, Ecuador, oangamarca@est.ups.edu.ec

R. Tufiño, Universidad Politécnica Salesiana, Quito, Ecuador, rtufino@ups.edu.ec

H. Ortega, Universidad Politécnica Salesiana, Quito, Ecuador, hortega@est.ups.edu.ec

² A. Zambrano, Escuela Politécnica Nacional, Quito, Ecuador, ana.zambrano@epn.edu.ec

programación C++. Además del uso de *Arch Linux* [12] como sistema operativo.

Este artículo se encuentra realizado bajo la siguiente estructura: En el primer punto detalla la introducción haciendo énfasis en el planteamiento del problema; siguiente, materiales y métodos del sistema; el tercer punto correspondiente a resultados y discusión del sistema desarrollado; y por último, las conclusiones y trabajo futuro son presentados.

II. MATERIALES Y MÉTODOS

El diseño se compone de dos módulos. El primero radica en el tratamiento y segmentación de la carta obtenida de la captura de una cámara web en el prototipo adaptado. El segundo módulo, enfocado en una RNA capaz de aprender e identificar diferentes tipos de caracteres (A, 2, 3, 4, 5, 6, 7, J, Q, K) correspondientes a cada carta mediante una etapa de entrenamiento.

Los elementos usados para la implementación del sistema con el uso de la VA e IA se muestran en la Fig. 1, donde la imagen es capturada a través de una cámara web con la iluminación adecuada, y pasa a ser analizada y procesada por algoritmos de VA dentro de un computador central para dar como resultado final una salida de audio que describe el caracter de cada carta.



Figura 1. Estructura el sistema de reconocimiento de cartas.

A. Modulo 1: Tratamiento y segmentacion de imagen

Para el funcionamiento del primer módulo, se realiza la identificación de la imagen, para obtener los atributos de cada carta. En la Fig. 2 se define la metodología por etapa.

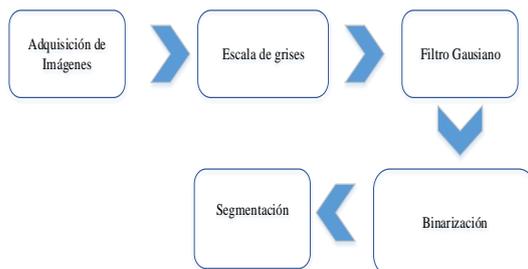


Figura 2. Etapas a seguir para el tratamiento de la imagen.

1) Adquisición de imágenes

En esta etapa se captura la imagen, es decir, se transmite información del mundo físico hasta la memoria del computador, a través de una cámara web estándar con resolución de 5 MP y precisión en el enfoque.

El fondo de la escena es de color negro, puesto que disipa los efectos lumínicos que presentan los colores claros teniendo en cuenta, que el color de las cartas en su mayoría es blanco. La cámara se ubica a 17 cm de distancia de la escena para visualizar el entorno de las 5 cartas, así como un foco led de 3W y 180 Lm, como se muestra en la Fig. 3. Un ejemplo de la imagen tomada por la cámara en tiempo de ejecución del sistema se visualiza en la Fig. 4.



Figura 3. Dispositivo de reconocimiento automático de cartas.

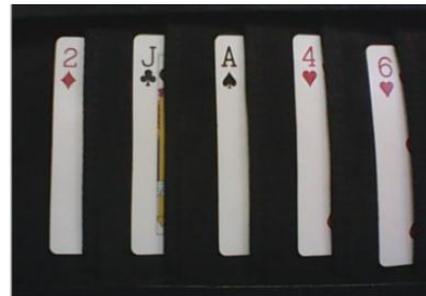


Figura 4. Imagen real capturada por la cámara.

2) Escala de Grises

El siguiente paso es convertir la imagen digital capturada a un formato adecuado para extraer sus características; cada pixel de la imagen se representa con valores entre 0 y 255, que indicará los diferentes niveles de grises. Para realizar esta conversión a cada pixel se hace un promedio equilibrado de la intensidad de cada uno de los tres colores del modelo RGB (Red/Green/Blue), donde a cada color se le asigna un peso. Al ser un promedio equilibrado la suma de los tres pesos debe ser igual a 1. *OpenCv* utiliza los siguientes pesos [13]. La ecuación (1) aplicada a la imagen real capturada da como resultado la Fig. 5:

$$Gris = (0,2989 * rojo) + (0,5870 * verde) + (0,114 * azul) \quad (1)$$

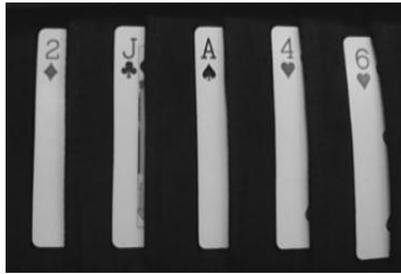


Figura 5. Imagen en escala de grises.

3) Filtro

El ruido en una imagen digital siempre existe, ya que ciertos pixeles tienen valores alterados y por tanto muestran distintos niveles de grises con respecto a sus pixeles vecinos. Para excluir este ruido, se reduce la amplitud de las variaciones sustituyendo cada píxel por la media del valor de los píxeles de sus vecinos. Se usó un filtro Gaussiano cuyo algoritmo se observa en la siguiente ecuación [14]:

$$G(x, y) = \frac{1}{2 * \pi * \sigma^2} * e^{-\frac{x^2 + y^2}{2 * \sigma^2}} \quad (2)$$

Donde: “x” es la distancia desde el origen en el eje horizontal, “y” es la distancia en el eje vertical y “σ” es la desviación propia de la distribución Gaussiana.

Esta ecuación (2) genera una superficie de contornos con círculos concéntricos, como se muestra en la Fig. 6.

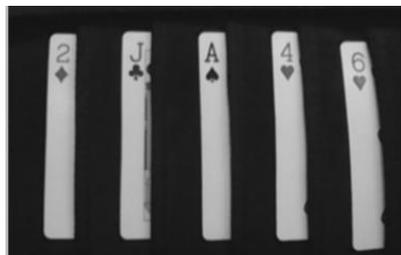


Figura 6. Imagen aplicada filtro Gaussiano.

4) Binarización

A continuación se transforma la imagen de varios niveles de grises a una nueva con solo dos valores, dependiendo si supera o no un umbral a definir. Todos los valores superiores al valor umbral serán binarizados a 1 y los que no superen ese valor lo serán a 0. Un método muy utilizado para binarizar una imagen es escoger un valor umbral $T(x, y)$ para cada píxel (x, y) y establecer el valor de cada píxel con la función *THRESH_BINARY* [15]:

$$dst(x, y) = \begin{cases} 0 & \text{si } I(x, y) < T(x, y) \\ 255 & \text{en cualquier otro caso} \end{cases} \quad (3)$$

Con la ecuación (3) se determina si el píxel está por abajo o por arriba del umbral. Se utilizó el método de

Otsu [16] [17] para determinar el nivel óptimo de umbral, ya que si se escoge un valor de umbral fijo se puede perder pixeles debido a la variación de luz. El resultado obtenido al aplicar las funciones *THRESH_BINARY* y *THRESH_OTSU* es el que se muestra en la Fig. 7.

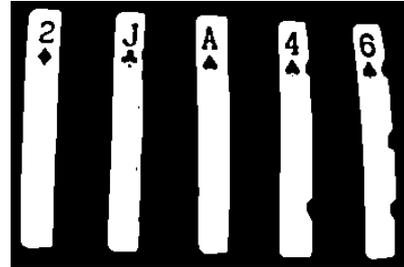


Figura 7. Imagen binarizada.

5) Segmentación

En esta etapa la imagen digital se subdivide en partes con significado, es decir, se separan las secciones de interés del resto de la imagen para realizar la extracción de información. Para la segmentación de imágenes se usa el método de detección de contornos aplicando el algoritmo propuesto por Suzuki y Abe [18]. Este algoritmo barre la imagen buscando píxel a píxel con el mismo valor y determinando cuáles de ellos pueden pertenecer a un mismo borde como se muestra en la Fig. 8.

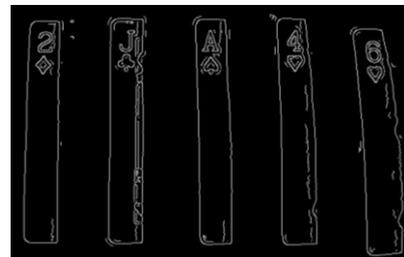


Figura 8. Segmentación por detección de contornos.

Posterior todos los contornos encontrados son almacenados y ordenados de forma secuencial. Se rellena de blanco los contornos encontrados, obteniendo como resultado la Fig. 9.



Figura 9. Relleno de contornos.

Se verifica la secuencia de datos almacenados anteriormente para separar los contornos de las áreas de mayor tamaño. A todos los contornos encontrados se enmarca, devolviendo el ancho, la altura y el

punto correspondiente a la esquina izquierda del rectángulo. Se guardan todos los valores en un arreglo, con el fin de utilizarlos posteriormente para calcular el área del rectángulo, cuyo valor refleja el tamaño y forma del contorno. A continuación se realiza una comparación de las áreas encontradas, donde el área de interés depende de la distancia a la cual se encuentre ubicada la cámara web (para este proyecto esta distancia es fija de 17cm debido al prototipo diseñado). Por este motivo el área de interés debe ser mayor a 200 pixeles, tener un ancho mayor a 8 pixeles y una altura menor a 27 pixeles. En la Fig. 10 por ejemplo, el único contorno que cumple estas características es el contorno (a) de acuerdo con la Tabla I.

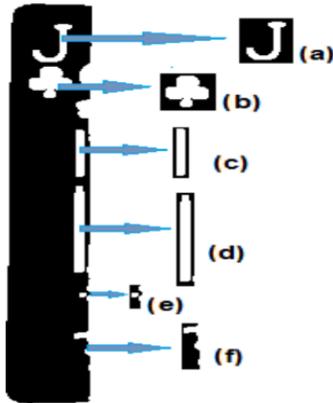


Figura 10. Ejemplo de segmentación para la carta J.

TABLA I. VALORES DE CONTORNOS PARA LA CARTA J.

Contorno	Ancho (Px)	Altura (Px)	Área (Px)	OK
(a)	20	24	480	Sí
(b)	15	13	195	No
(c)	4	22	88	No
(d)	5	47	235	No
(e)	4	9	36	No
(f)	7	23	161	No

El fin de este procedimiento es identificar las cinco cartas y ubicarlas de izquierda a derecha para ser localizadas por el jugador [19], una vez ordenados los contornos de las 5 cartas. Se aplica la ecuación (4) al área segmentada para invertir el color [15] y comparar con el conjunto de muestras del entrenamiento de la red. La Fig. 11 muestra el resultado de este procedimiento.

$$dst(x, y) = \begin{cases} 255 & \text{si } I(x, y) < T(x, y) \\ 0 & \text{en cualquier otro caso} \end{cases} \quad (4)$$



Figura 11. Área de interés segmentada e invertida.

B. Módulo 2: Reconocimiento de caracteres mediante una Red Neuronal Artificial

Este segundo módulo corresponde a la identificación del carácter de las cartas mediante las etapas como se muestra en la Fig. 12.

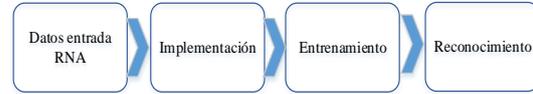


Figura 12. Módulo 2, Reconocimiento de caracteres usando una RNA.

1) Datos de entrada a la RNA

Para que la información de cada carácter pueda ser procesada por la RNA, es necesario convertir la imagen segmentada en una estructura de datos que permita representar los caracteres que se desea reconocer. En la Fig. 13 se representa los caracteres en una matriz de 16 x 16 pixeles. Los datos de entrada de la RNA deben ser representados por un vector lineal, cuyo tamaño será de $x = [x1, x2, x3, \dots, x256]$. Los elementos para el ejemplo de la carta "J" se muestran en la Fig. 14.

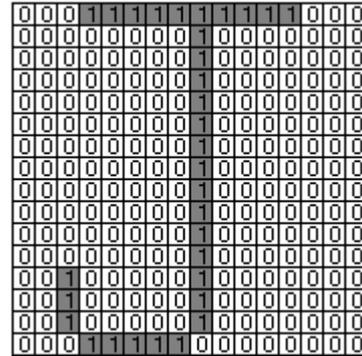


Figura 13. Carácter "J" representado en una matriz 16x16.

X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	X ₁₁	X ₁₂	X ₁₃	X ₁₄	X ₁₅	X ₁₆
X ₁₇	X ₁₈	X ₁₉	X ₂₀	X ₂₁	X ₂₂	X ₂₃	X ₂₄	X ₂₅	X ₂₆	X ₂₇	X ₂₈	X ₂₉	X ₃₀	X ₃₁	X ₃₂
X ₃₃	X ₃₄	X ₃₅	X ₃₆	X ₃₇	X ₃₈	X ₃₉	X ₄₀	X ₄₁	X ₄₂	X ₄₃	X ₄₄	X ₄₅	X ₄₆	X ₄₇	X ₄₈
X ₄₉	X ₅₀	X ₅₁	X ₅₂	X ₅₃	X ₅₄	X ₅₅	X ₅₆	X ₅₇	X ₅₈	X ₅₉	X ₆₀	X ₆₁	X ₆₂	X ₆₃	X ₆₄
X ₆₅	X ₆₆	X ₆₇	X ₆₈	X ₆₉	X ₇₀	X ₇₁	X ₇₂	X ₇₃	X ₇₄	X ₇₅	X ₇₆	X ₇₇	X ₇₈	X ₇₉	X ₈₀
X ₈₁	X ₈₂	X ₈₃	X ₈₄	X ₈₅	X ₈₆	X ₈₇	X ₈₈	X ₈₉	X ₉₀	X ₉₁	X ₉₂	X ₉₃	X ₉₄	X ₉₅	X ₉₆
X ₉₇	X ₉₈	X ₉₉	X ₁₀₀	X ₁₀₁	X ₁₀₂	X ₁₀₃	X ₁₀₄	X ₁₀₅	X ₁₀₆	X ₁₀₇	X ₁₀₈	X ₁₀₉	X ₁₁₀	X ₁₁₁	X ₁₁₂
X ₁₁₃	X ₁₁₄	X ₁₁₅	X ₁₁₆	X ₁₁₇	X ₁₁₈	X ₁₁₉	X ₁₂₀	X ₁₂₁	X ₁₂₂	X ₁₂₃	X ₁₂₄	X ₁₂₅	X ₁₂₆	X ₁₂₇	X ₁₂₈
X ₁₂₉	X ₁₃₀	X ₁₃₁	X ₁₃₂	X ₁₃₃	X ₁₃₄	X ₁₃₅	X ₁₃₆	X ₁₃₇	X ₁₃₈	X ₁₃₉	X ₁₄₀	X ₁₄₁	X ₁₄₂	X ₁₄₃	X ₁₄₄
X ₁₄₅	X ₁₄₆	X ₁₄₇	X ₁₄₈	X ₁₄₉	X ₁₅₀	X ₁₅₁	X ₁₅₂	X ₁₅₃	X ₁₅₄	X ₁₅₅	X ₁₅₆	X ₁₅₇	X ₁₅₈	X ₁₅₉	X ₁₆₀
X ₁₆₁	X ₁₆₂	X ₁₆₃	X ₁₆₄	X ₁₆₅	X ₁₆₆	X ₁₆₇	X ₁₆₈	X ₁₆₉	X ₁₇₀	X ₁₇₁	X ₁₇₂	X ₁₇₃	X ₁₇₄	X ₁₇₅	X ₁₇₆
X ₁₇₇	X ₁₇₈	X ₁₇₉	X ₁₈₀	X ₁₈₁	X ₁₈₂	X ₁₈₃	X ₁₈₄	X ₁₈₅	X ₁₈₆	X ₁₈₇	X ₁₈₈	X ₁₈₉	X ₁₉₀	X ₁₉₁	X ₁₉₂
X ₁₉₃	X ₁₉₄	X ₁₉₅	X ₁₉₆	X ₁₉₇	X ₁₉₈	X ₁₉₉	X ₂₀₀	X ₂₀₁	X ₂₀₂	X ₂₀₃	X ₂₀₄	X ₂₀₅	X ₂₀₆	X ₂₀₇	X ₂₀₈
X ₂₀₉	X ₂₁₀	X ₂₁₁	X ₂₁₂	X ₂₁₃	X ₂₁₄	X ₂₁₅	X ₂₁₆	X ₂₁₇	X ₂₁₈	X ₂₁₉	X ₂₂₀	X ₂₂₁	X ₂₂₂	X ₂₂₃	X ₂₂₄
X ₂₂₅	X ₂₂₆	X ₂₂₇	X ₂₂₈	X ₂₂₉	X ₂₃₀	X ₂₃₁	X ₂₃₂	X ₂₃₃	X ₂₃₄	X ₂₃₅	X ₂₃₆	X ₂₃₇	X ₂₃₈	X ₂₃₉	X ₂₄₀
X ₂₄₁	X ₂₄₂	X ₂₄₃	X ₂₄₄	X ₂₄₅	X ₂₄₆	X ₂₄₇	X ₂₄₈	X ₂₄₉	X ₂₅₀	X ₂₅₁	X ₂₅₂	X ₂₅₃	X ₂₅₄	X ₂₅₅	X ₂₅₆

Figura 14. Valor de cada pixel, para obtener al vector.

2) Implementación de la Red Neuronal Artificial

Por características de la investigación se decide implementar una RNA de tipo perceptrón multicapa ó *Multi Layer Perceptron (MLP)* por sus siglas en

inglés [20] [21], que trabaja con aprendizaje supervisado facilitando distinguir un conjunto de muestras logrando asociar los datos de entrada con su correspondiente conjunto de salida. Esta red MLP tiene como características: tolerante a fallas, rápida implementación y permite generalizar las respuestas frente a nuevos patrones de entrada basado en aproximación. El esquema de la red MLP se muestra en la Fig. 15.

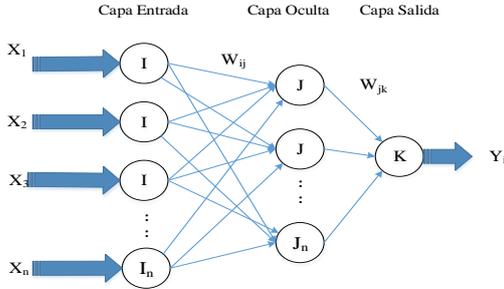


Figura 15. Esquema de un MLP.

Como se puede observar las unidades de procesos de entrada en cada capa, son las salidas de las unidades de proceso de la capa anterior. El MLP implementa en cada neurona la siguiente ecuación:

$$y_i = f \left(\sum_{n=1}^L x_n w_n \right) \quad (5)$$

Donde y_i : Salida; f : Función de activación en cada neurona; w_n : Peso entre las capas; y finalmente x_n : Datos de entradas.

La función de activación (f) es generalmente Sigmoidal [20], debido que es continua y fácilmente diferenciable con dos únicos valores 0 ó 1. Una vez definido el tipo de RNA y el tipo función de activación, se procede a determinar el número de neuronas necesarias para cada capa. La capa de entrada tiene 256 neuronas, puesto que, cada caracter a evaluar posee una dimensión de 16x16 píxeles, con un total de 256 datos de entrada unos (1) y ceros (0). Debido a que no existe ningún método para determinar el número de neuronas de la capa oculta, se realizan pruebas con diferentes neuronas para establecer el valor óptimo en cuanto a tiempo de aprendizaje y efectividad, dando como resultado 16 neuronas (ver resultados). Por otro lado la capa de salida contará con 10 neuronas, ya que es el número de cartas que se requiere reconocer.

3) Entrenamiento de una RNA para el reconocimiento de caracteres

El objetivo es enseñar a la red a distinguir los diferentes caracteres de todas las cartas. El entrenamiento inicia llamando a un conjunto de muestras compuesto de 330 imágenes por caracter,

tomando como base los distintos tipos de fuente. Por ejemplo para el caracter “J”, existen diferentes tipos de fuente, como se muestra en la Fig. 16. Este conjunto de muestras están organizadas por caracter, y serán invocadas desde el programa, para realizar el entrenamiento de la red. Mientras más imágenes se tengan mayor será el número de probabilidad de reconocimiento de las cartas.

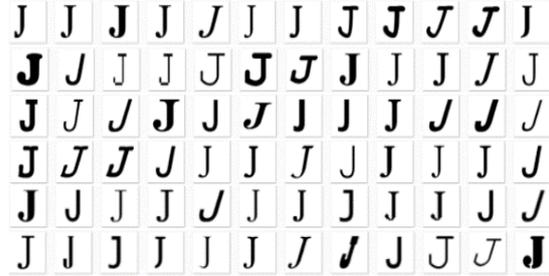


Figura 16. Tipos de representación del caracter “J”.

Para el entrenamiento de la RNA se utiliza el algoritmo de Retropropagación o también conocido como *Backpropagation* [22], permitiendo ajustar los pesos de forma que reduzca el error entre la salida deseada y la respuesta de la red. Haciendo referencia a la Fig. 15, se muestra un MLP donde se diferencia las tres capas usadas: entrada (i), oculta (j) y salida (k). Lo que se calcula primero es el error de la capa de salida (k),

$$E_k = (T_k - O_k) * O_k * (1 - O_k) \quad (6)$$

Donde: T_k es la salida correcta ó el valor que debe aprender la red y O_k es la salida actual.

Este error es calculado individualmente para las dos neuronas que se tiene en la capa de salida (k). A continuación se calculan los pesos entre las capas (j, k) con la siguiente fórmula:

$$W_{jk} = W_{jk} + L * E_k * O_j \quad (7)$$

Donde: w_{jk} es el peso entre las capas (j, k), L es el factor de aprendizaje y O_j la entrada a la capa K desde J . Luego se calcula el error en la capa oculta (j); que se calcula multiplicando la salida O_j por la sumatoria de los errores de la capa de salida (k) multiplicado por el peso (j, k).

$$E_j = O_j (1 - O_j) * \sum E_k * W_{jk} \quad (8)$$

Por último se determina los pesos entre la capa de entrada y la capa oculta (i, j):

$$W_{ij} = W_{ij} + L * E_j * O_i \quad (9)$$

Donde: w_{ij} es el peso entre las capas (i, j), L el factor de aprendizaje y O_j la entrada a la capa J .

Para entrenar la RNA se realizaron 1.000 iteraciones de entrenamiento; *OpenCV* tiene la opción de almacenar la configuración y los pesos (Parámetros.XML) de la RNA, en forma de archivo con extensión .XML.

4) Reconocimiento de patrones

El siguiente paso es cargar el archivo que tiene el modelo entrenado de la RNA, para inicializar la predicción. Se emplea un método de predicción en donde se ingresa el vector de entrada que estará representado por $x = [x_1, x_2, x_3 \dots x_{256}]$ y entrega un vector de salida representado por $y = [y_1, y_2, y_3, \dots, y_9, y_{10}]$ donde la RNA escribirá los resultados de la predicción.

Para concluir el proceso de reconocimiento de caracteres, el sistema compara uno por uno los valores resultantes de cada posición del vector de salida y , determinando así cual es la posición del máximo valor. Por ejemplo si el máximo valor se encuentra en la primera posición del vector y la carta reconocida será la carta "As". Si el máximo valor se encuentra en la segunda posición será la carta 2, y así sucesivamente. Los valores negativos o nulos son ignorados [23]. Como ejemplo para los vectores de entrada y salida de la carta "J" se muestran gráficamente en la siguiente Fig. 17:

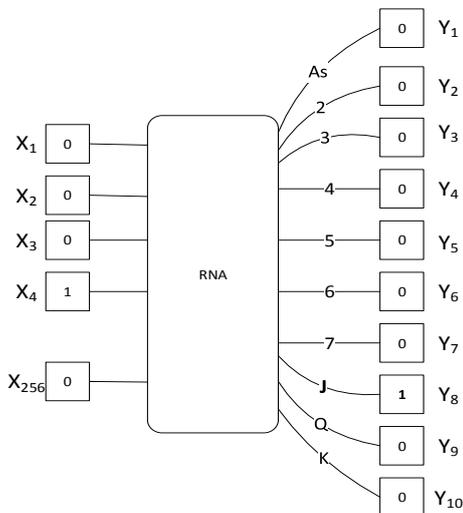


Figura 17. Representación de datos de entrada y salida para el reconocimiento del carácter "J".

El resultado final es una salida de audio que informa el carácter de cada carta de izquierda a derecha. Cada carta está asociada a un sonido con su respectivo valor, el mismo que se reproduce usando la librería *QMedia Player* de *OpenCV*. Este audio es escuchado por la persona no vidente por audífonos para que pueda seleccionar una carta en función de la necesidad del juego.

III. RESULTADOS Y DISCUSIÓN

Las gráficas y tablas presentadas en esta sección, muestran los resultados de las pruebas realizadas para determinar las capas ocultas en la RNA y la efectividad del reconocimiento.

A. Determinación de capas ocultas en la RNA

Cuanto mayor es el número de neuronas en la capa oculta, se determinó que el tiempo de aprendizaje es mayor, mientras que, con un número menor de neuronas los resultados presentan menor precisión. Los mejores resultados se obtuvieron al tener 16 neuronas en la capa oculta, conforme a las pruebas de entrenamiento realizadas como muestran las pruebas realizadas en la Fig. 18 y 19. Se puede apreciar que a partir de 16 neuronas el aumento de efectividad es mínimo y por otra parte al utilizar más de 16 neuronas el tiempo de aprendizaje aumenta.

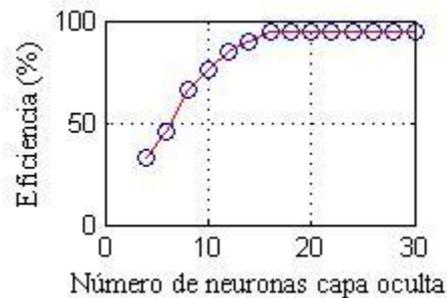


Figura 18. Número de neuronas capa oculta vs. Efectividad [%].

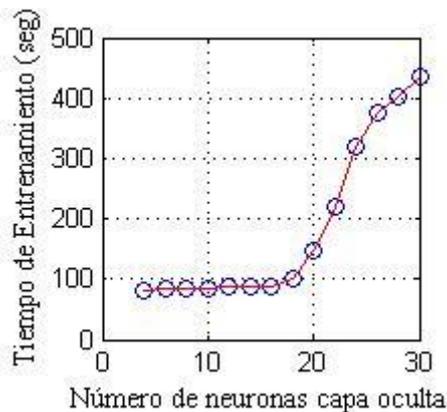


Figura 19. Número de neuronas capa oculta vs. Tiempo de entrenamiento [segundos].

B. Precisión del reconocimiento

La iluminación es fundamental, puesto que dependiendo de la ubicación puede ocultar o destacar atributos de las cartas, creando tipos de datos confiables a ser procesadas.

La precisión del reconocimiento de las cartas se analiza a partir de una matriz de confusión que permite validar la exactitud del reconocimiento [24]. Estas pruebas se muestran en la Tabla II y Tabla III, donde se realizaron 100 pruebas por cada carta para

probar el sistema con luz y sin luz artificial. Se puede observar en la Tabla II, que las cartas con caracteres 3 y “Q” tienen un porcentaje bajo de efectividad de reconocimiento; de las 100 pruebas realizadas solamente 66 y 79 veces respectivamente se reconoció correctamente. En base a las pruebas realizadas se determinó que la cantidad de Luxes recomendado para el adecuado funcionamiento del sistema es de 241 ± 25 Luxes. Para esto se implementó un sistema de iluminación artificial empleando un foco led a una distancia de 16cm de

las cartas para mitigar la sombra que produce las hendiduras del estuche y la luz ambiente natural. Usando el sistema de iluminación artificial el reconocimiento de la carta 3 mejoró de un 66% a un 87% y de la carta “Q” mejoró de un 79% a un 95%, haciendo que la fiabilidad global del sistema incremente a 94,70%. En la Fig. 20 se muestra gráficamente los resultados obtenidos luego de realizar las pruebas del sistema de reconocimiento con y sin luz para cada carta.

TABLA II. MATRIZ DE CONFUSIÓN SIN LUZ ARTIFICIAL.

		Cartas Reales										Total Reconocidas	Error de comisión (%)	Exactitud con la referencia (%)
		A	2	3	4	5	6	7	J	Q	K			
Cartas Reconocidas	A	97	0	0	7	0	0	0	0	0	3	107	9,35	90,65
	2	0	90	0	0	4	0	8	4	8	0	114	21,05	78,95
	3	0	0	66	0	2	3	0	0	0	0	71	7,04	92,96
	4	3	0	4	87	0	0	2	0	5	6	107	18,69	81,31
	5	0	0	7	0	86	4	0	0	0	3	100	14,00	86,00
	6	0	5	0	0	0	86	0	0	1	0	92	6,52	93,48
	7	0	3	6	6	0	0	86	2	3	8	114	24,56	75,44
	J	0	0	17	0	0	0	4	94	4	0	119	21,01	78,99
	Q	0	2	0	0	0	7	0	0	79	0	88	10,23	89,77
	K	0	0	0	0	8	0	0	0	0	80	88	9,09	90,91
	Total Real	100	100	100	100	100	100	100	100	100	100	Suma Diagonal Principal:		851
Error de omisión (%)	3,00	10,00	34,00	13,00	14,00	14,00	14,00	6,00	21,00	20,00	Total Cartas Reales:		1000	
Exactitud del sistema (%)	97,00	90,00	66,00	87,00	86,00	86,00	86,00	94,00	79,00	80,00	Fiabilidad Global (%):		85,10	

*No se toma en cuenta las cartas 8, 9 y 10 por reglas del juego.

TABLA III. PRUEBA CON LUZ ARTIFICIAL.

		Cartas Reales										Total Reconocidas	Error de comisión (%)	Exactitud con la referencia (%)
		A	2	3	4	5	6	7	J	Q	K			
Cartas Reconocidas	A	98	0	0	3	0	0	0	0	0	1	102	3,92	96,08
	2	0	98	0	0	0	0	3	0	0	0	101	2,97	97,03
	3	0	0	87	0	0	1	0	0	0	0	88	1,14	98,86
	4	1	0	1	95	0	0	0	0	2	2	101	5,94	94,06
	5	0	0	5	0	96	4	0	1	0	1	107	10,28	89,72
	6	0	2	2	0	1	92	2	0	1	0	100	8,00	92,00
	7	1	0	3	1	1	0	94	1	1	2	104	9,62	90,38
	J	0	0	2	1	0	0	0	98	1	0	102	3,92	96,08
	Q	0	0	0	0	0	3	0	0	95	0	98	3,06	96,94
	K	0	0	0	0	2	0	1	0	0	94	97	3,09	96,91
	Total Real	100	100	100	100	100	100	100	100	100	100	Suma Diagonal Principal:		947
Error de omisión (%)	2,00	2,00	13,00	5,00	4,00	8,00	6,00	2,00	5,00	6,00	Total Cartas Reales:		1000	
Exactitud del sistema (%)	98,00	98,00	87,00	95,00	96,00	92,00	94,00	98,00	95,00	94,00	Fiabilidad Global (%):		94,70	

*No se toma en cuenta las cartas 8, 9 y 10 por reglas del juego.

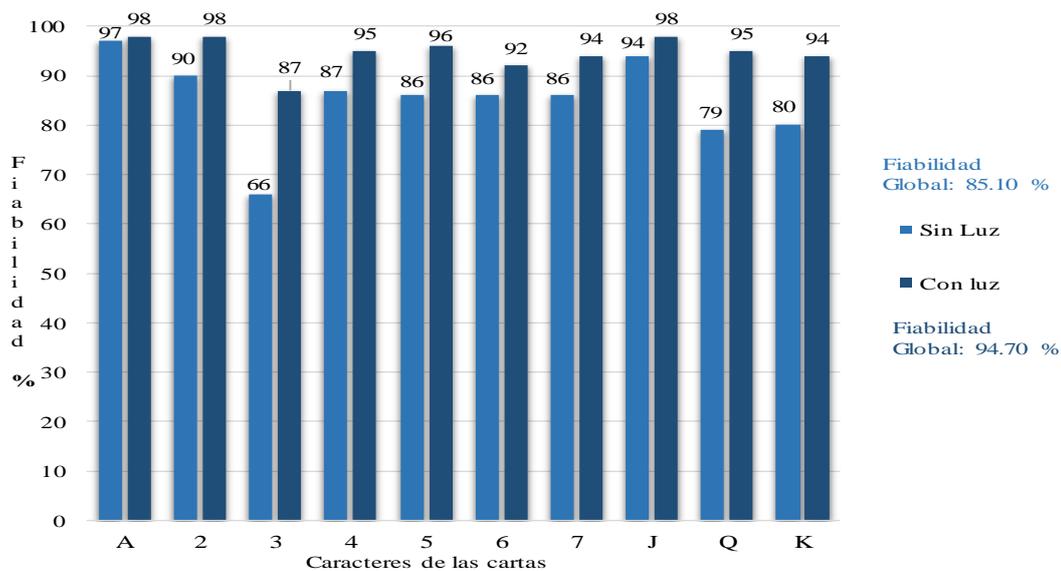


Figura 20. Histograma de efectividad para cada carta.

IV CONCLUSIONES Y TRABAJO FUTURO

Los algoritmos de Visión e Inteligencia Artificial aportan a solucionar en gran medida el problema específico de inclusión social planteado en el presente estudio, dirigido a personas no videntes a ser partícipes del juego de mesa popular en el Ecuador llamado 40. Definiendo un proceso para la adquisición, identificación y reconocimiento del carácter de cada carta. Logrando obtener un margen de error del 5,3% en el sistema, este porcentaje se determinó al realizar 100 pruebas del reconocimiento de cada carta, permitiendo obtener los resultados deseados.

El sistema implementado obtuvo un 85,10% de efectividad para reconocer las diez cartas del juego, incrementó al 94,70%, gracias a las pruebas posteriores que se realizaron con luz artificial. Como se observó, uno de los inconvenientes a la hora de resolver el problema del reconocimiento de las cartas es la intensidad de luz ambiente, ya que son factores externos muy variables. Para corregirlo se implementó un mecanismo que mantiene la intensidad de luz fija, con esto se observa que el sistema es capaz de reconocer las 10 cartas.

La elección y correcta ubicación de los elementos del prototipo como: el color del soporte de las cartas, fuente de iluminación y la distancia que se ubicó la cámara, redujeron considerablemente el número de algoritmos de Visión Artificial necesarios para la elaboración del sistema, logrando así alcanzar los objetivos planteados.

Una vez que las cartas han sido reconocidas, la salida de audio debe tener una pausa entre cada reproducción, ya que en las primeras pruebas realizadas no se entendía que cartas fueron reconocidas, el tiempo estimado para una buena

compresión es de 2 segundos, esto dará tiempo a que el jugador memorice las cartas que le fueron repartidas.

El actual proyecto de investigación será el inicio para el desarrollo de un proyecto más grande que pretende aportar mediante la creación de un dispositivo que permitan jugar completamente a dos participantes no videntes el juego de "40".

REFERENCIAS

- [1] L. Salgado, "Visión Artificial: Fundamentos y Aplicaciones," em VA: *Fundamentos y Aplicaciones*, Madrid, 2007.
- [2] J. O. M. a. J. R. A. Castillo, "Virtual Laboratory for Digital," *IEEE LATIN AMERICA TRANSACTIONS*, vol. 12, n° 6, p. 1176, 2014.
- [3] J. d. A. Suárez, "Técnicas de inteligencia artificial aplicadas al análisis de la solvencia empresarial," 2012. [Online]. Disponible: <https://dialnet.unirioja.es/servlet/articulo?codigo=1252835>. [Acceso em 2016].
- [4] I. N. d. E. y. Censos, "http://www.ecuadorencifras.gob.ec/," 2010. [Online]. Disponible: http://www.inec.gob.ec/cpv/index.php?option=com_content&view=article&id=232&Itemid=128&lang=es. [Acceso en 2016].
- [5] J. F. Parra Buestan, "Repositorio Institucional Universidad de Cuenca," 2010. [Online]. Disponible: <http://dspace.ucuenca.edu.ec/handle/123456789/1726>. [Acceso en 2016].
- [6] Nora La Serna Palomino, Ulises Román Concha, "Técnicas de segmentación en procesamiento digital de imágenes," *Revista de Ingeniería de Sistemas e Informática*, vol. 6, n° 2, 2009.
- [7] I. L. J. S. M. P. V. L. a. G. L. R. Osorio, "Surveillance System Mobile Object Using," *IEEE LATIN AMERICA TRANSACTIONS*, vol. 13, n° 7, p. 2441, 2015.
- [8] X. B. Olabe, "REDES NEURONALES ARTIFICIALES Y

- SUS APLICACIONES,” 2014. [Online]. Disponible: http://www.ciberesquina.una.edu.ve:8080/2014_2/350_E.pdf. [Acceso en 2016].
- [9] Fernando Izaurieta, Carlos Saavedra, “Redes Neuronales Artificiales,” em *Redes Neuronales Artificiales*, Concepción.
- [10] A. Tierra, “Prediction 3-D Velocity for Ecuador by,” *IEEE LATIN AMERICA TRANSACTIONS*, vol. 14, nº 1, p. 386, 2016.
- [11] Opencv dev team, “Open CV,” 2014. [Online]. Disponible: <http://docs.opencv.org/2.4.10/>.
- [12] L. T. Inc., “The Arch Linux,” 2002. [Online]. Disponible: <https://www.archlinux.org/>. [Acceso en 2016].
- [13] S. Gómez González e S. Gutiérrez Alzate, “<http://repositorio.utp.edu.co/dspace/>,” 2011. [Online]. Disponible: <http://repositorio.utp.edu.co/dspace/handle/11059/2458>. [Acceso en 2016].
- [14] F. A. V. Á. Carlos Alberto Júnez Ferreira, “Algoritmos para la reducción de ruido en imágenes digitales,” *Facultad de Ingeniería Civil*, nº 2, p. 23, 2012.
- [15] P. P. G. GARCÍA, “RECONOCIMIENTO DE IMÁGENES UTILIZANDO REDES NEURONALES ARTIFICIALES,” 2013. [Online]. Disponible: <http://eprints.ucm.es/23444/>.
- [16] Universidad Nacional de Quilmes, “Segmentación por Umbralización,” 2005. [Online]. Disponible: <http://iacci.unq.edu.ar/materias/vision/apuntes.htm>. [Acceso em 2016].
- [17] M. S. S. A. V. L. a. D. M. R. C. Marsal, “Morphological And Statistical Analysis Of Biomaterials With Applications In Tissue Engineering By Means Of Microscopy Image Processing,” *IEEE LATIN AMERICA TRANSACTIONS*, vol. 9, nº 3, p. 406, 2011.
- [18] Satoshi Suzuki, Keiichi Abe, “Topological Structural Analysis of Digital Binary Images by BorderFollowing,” *CVGIP*, nº 1, 16 Diciembre 1983.
- [19] O. A. A. Aquino, “Reo Cities,” 20 12 2005. [Online]. Disponible: <http://reocities.com/octavioalberto.geo/math/ordenamiento.pdf>. [Acceso en 2016].
- [20] M. Vidal González, “Repositorio de la Universidad de las Illes Balears,” 01 07 2015. [Online]. Disponible: <http://repositori.uib.es/xmlui/handle/11201/1126>. [Acceso em 2016].
- [21] G. R. L. G. J. R. M. a. M. T. P. L. M. L. Orozco, “Identification Inverted Pendulum System Using,” *IEEE LATIN AMERICA TRANSACTIONS*, vol. 13, nº 5, p. 1569, 2015.
- [22] G. Colmenares, “<http://webdelprofesor.ula.ve/index.php>,” 05 12 2012. [Online]. Disponible: http://webdelprofesor.ula.ve/economia/gcolmen/programa/economia/backpropagation_ma.pdf. [Acceso en 2016].
- [23] A. C. V. A. M. H. L. A. L. P. R. N. Hugo Vega Huerta, “Revista de investigación de Sistemas e Informática,” 2 Julio 2009. [Online]. Disponible: <http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3301>. [Acceso en 2016].
- [24] I. ARG, “<http://www.ign.gob.ar/descargas/CursoIso/>,” 2010. [Online]. Disponible: <http://www.ign.gob.ar/descargas/CursoIso/MODULO3-EJEMPLO-APLIC.pdf>. [Acceso en 2016].
- [25] Stuart J. Russell ; Peter Norving, *Inteligencia artificial un enfoque moderno*, Madrid: Pearson Educación S.A., 2004.
- [26] P. P. Cruz, *Inteligencia artificial con aplicaciones a la ingeniería*, México D.F.: Alfaomega, 2010.
- [27] S. N. C. Hernández, “Repositorio digital de Tesis PUCP,” Junio 2014. [Online]. Disponible: <http://tesis.pucp.edu.pe/repositorio/handle/123456789/5956>.
- [28] Patricia Nataly Constante Prócel; Andrés Marcelo Gordón Garcés, “Repositorio Digital EPN,” Agosto 2015. [Online]. Disponible: <http://bibdigital.epn.edu.ec/handle/15000/11368>.
- [29] D. R. G. Chunhui (Brenda) Zheng, “CiteSeer,” 2007. [Online]. Disponible: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.329.9225&rank=1>. [Acceso en 2016].
- [30] H. W. Dan Brinks, “Texas Hold'em Hand Recognition and Analysis,” *Stanford Digital Repository*, p. 6, 2007.

ANEXOS

Anexo 1. Características técnicas cámara



ESPECIFICACIONES ▲

Requisitos del sistema

1 GHz (se recomiendan 1,6 GHz)
512 MB de RAM o más
200 MB de espacio libre en el disco duro
Conexión a Internet
Puerto USB 1.1 (se recomienda 2.0)

Para disfrutar de las mejores videoconferencias, sugerimos:

CPU con doble núcleo y 1 GB de RAM
Conexión a Internet por banda ancha con carga de 256 kbps o superior

Información de garantía

Dos años de garantía de hardware limitada

Esta caja contiene

Cámara web con cable USB
Documentación del usuario

Especificaciones

Videoconferencias (640 x 480 píxeles) con el sistema recomendado
Captura de vídeo: Hasta 1024 x 768 píxeles
Tecnología Logitech Fluid Crystal™
Fotos: Hasta 5 megapíxeles (mejora por software)
Micrófono integrado con reducción de ruido
Certificación USB 2.0 de alta velocidad (se recomienda)
Clip universal para monitores LCD, CRT o portátiles

Software de cámara Web Logitech:
Controles de panorámico, inclinación y zoom
Captura de vídeo y fotos
Seguimiento facial
Detección de movimiento

Anexo 2. Características técnicas mini CPU



Intel® NUC Kit DN2820FYKH TECHNICAL SPECIFICATIONS

PROCESSOR

- Intel® Celeron® Processor N2820 (up to 2.39 GHz, dual-core processor with 1MB cache)
- Supports Intel® Smart Connect Technology³
- Supports Intel® Virtualization Technology (VT-x)
- Supports Intel® 64 architecture⁴

GRAPHICS

- Intel® HD Graphics (up to 756 MHz Graphics Max Dynamic Frequency)
- One HDMI* 1.4a port

SYSTEM MEMORY

- One DDR3L SO-DIMM slot for memory expandability up to 8 GB
- 1.35V, 1333/1600 MHz (down clocked to 1066 MHz)

PERIPHERAL CONNECTIVITY

- Intel® Wireless-N 7260BN PCIe* mini card pre-installed (802.11bgn, Bluetooth* 4.0, Intel® Wireless Display)
- Integrated 10/100/1000 Mbps Ethernet Network Connection
- One SuperSpeed USB 3.0 port (front panel)
- Two Hi-Speed USB 2.0 ports (back panel)

SYSTEM BIOS

- 64 Mb Flash EEPROM with Intel® Platform Innovation Framework for EFI Plug and Play
- Advanced configuration and power interface V3.0b, SMBIOS2.5
- Intel® Visual BIOS
- Intel® Visual Boot Manager
- Intel® Express BIOS update support
- Optimized POST for almost instant-on access to PC from power on

HARDWARE MANAGEMENT FEATURES

- Processor fan speed control
- Voltage and temperature sensing
- Fan sensor inputs used to monitor fan activity
- ACPI-compliant power management control

EXPANSION CAPABILITIES

- One PCI Express* half-length mini-card connector⁵ (factory pre-installed with wireless mini-card)
- 1x SATA port (3 Gb/s) for internal support for 2.5" HDD or SSD

AUDIO

- Intel® HD Audio⁶ via HDMI 1.4a output supporting 8 channel (7.1) digital audio
- Intel® HD Audio via Realtek ALC283 and 1/8" analog audio jack, supporting headset, speakers, headphones, or microphone

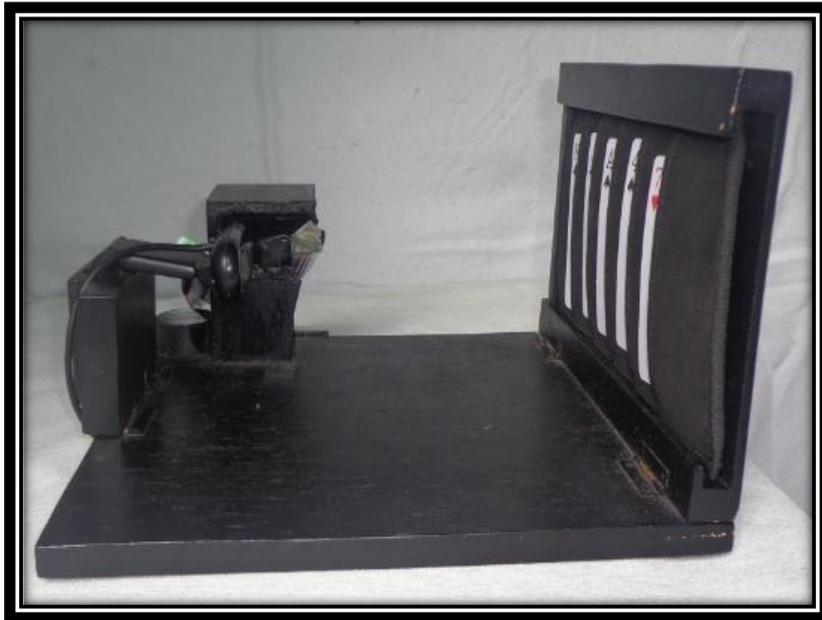
Anexo 3. Fotografías del Prototipo



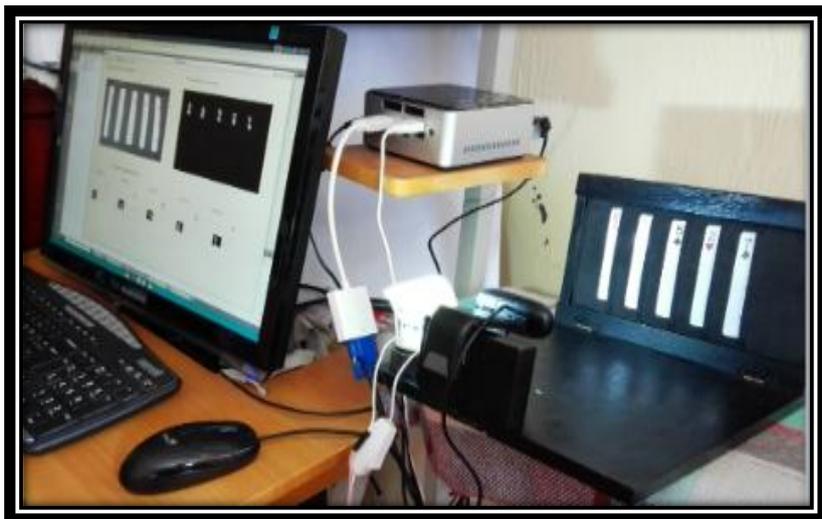
Vista lateral mostrando los bordes del estuche



Vista frontal con iluminación artificial

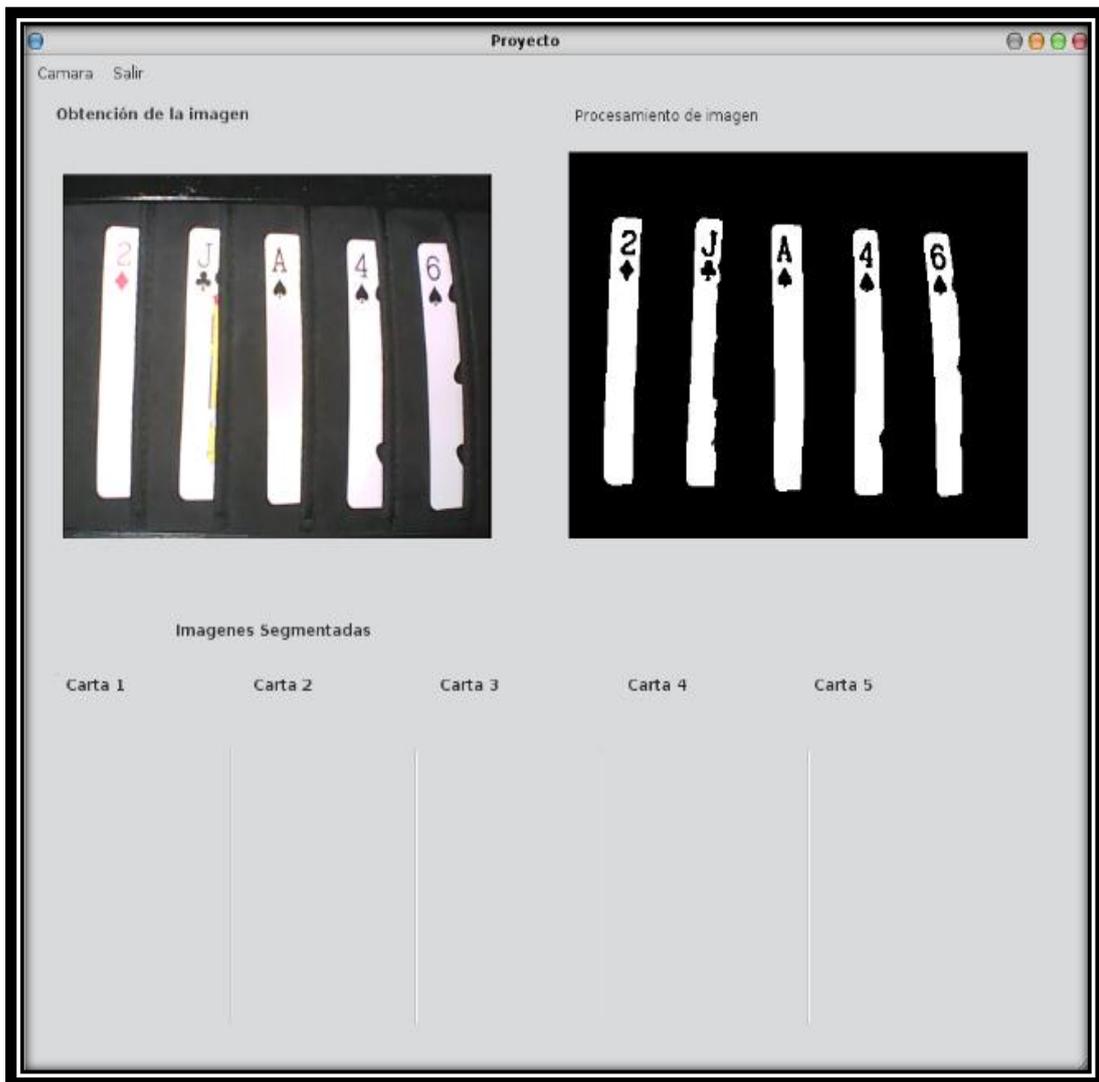


Vista frontal con iluminación artificial

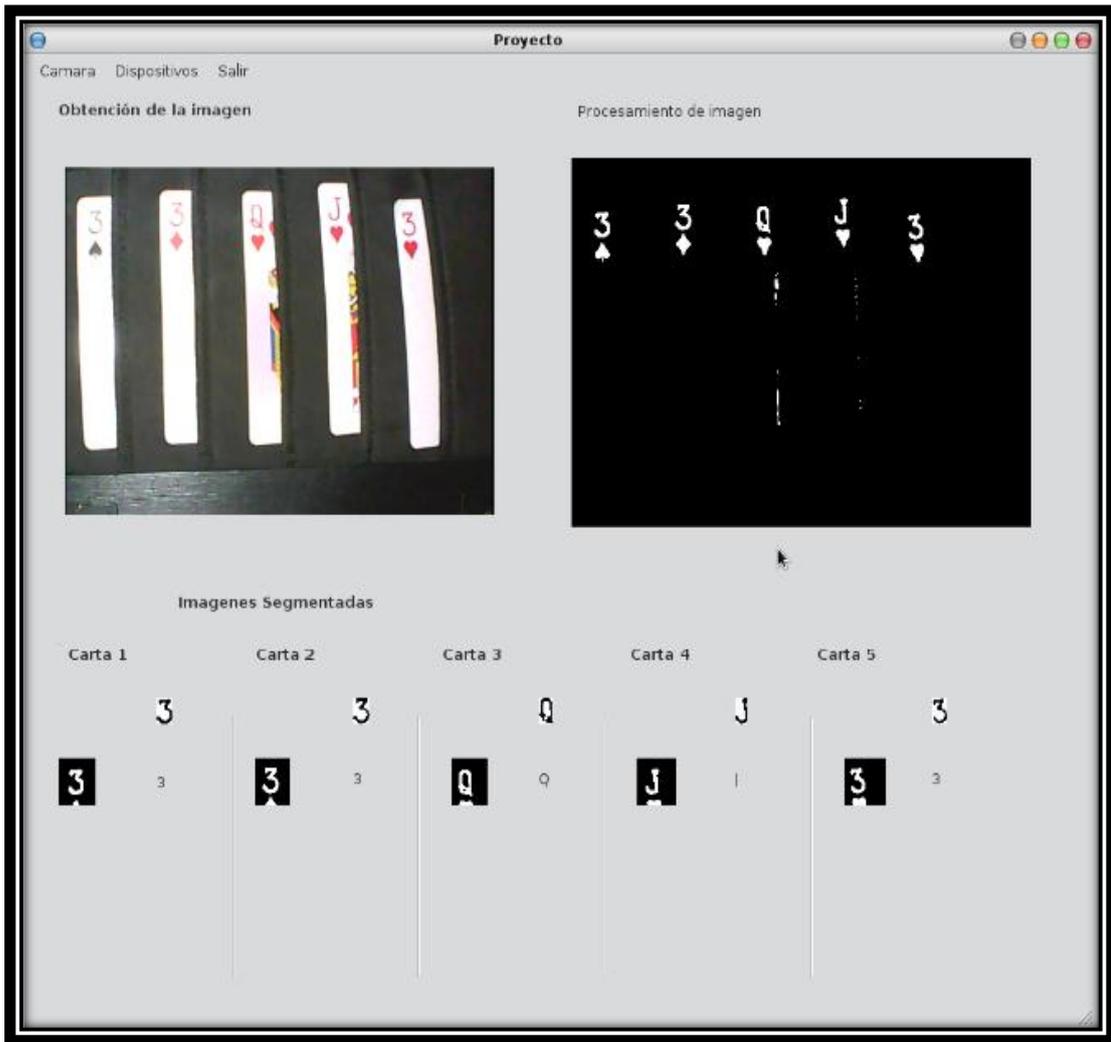


Vista general del sistema

Anexo 4. Interfaz gráfica



Pantalla Principal: Imagen capturada



Pantalla Principal: Imagen procesada