



SISTEMA DE RECONOCIMIENTO DE PATRONES VISUALES BASADO EN T CNICAS DE PROCESAMIENTO DE IM GENES Y REDES NEURONALES

Wai O,ChoTse
Universidad Rafael Belloso Chac n. Venezuela

RESUMEN

Esta investigaci n tuvo como finalidad el desarrollo de un sistema de reconocimiento de patrones visuales basado en t cnicas de procesamiento de im genes y redes neuronales. El estudio est  orientado al dise o y construcci n del sistema, consider ndose una investigaci n de tipo proyectiva, a su vez seg n su m todo es descriptivo y por su fuente es documental. La metodolog a utilizada es propia de Cho (2002), desarrollada sobre las bases de Angulo (1996). El software del sistema est  desarrollado en el ambiente de programaci n Matlab 6.5, y el algoritmo utiliza los principios de procesamiento de im genes para segmentar y filtrar la foto, para finalmente aplicar la teor a de redes neuronales LVQ para el reconocimiento. El hardware est  construido por piezas de maderas, los actuadores son de tipo pist n, y utiliza una Webcam como dispositivo de captura de imagen. Adicionalmente, se dise o y se construy  la interfaz basada en rel s y optoacopladores que conecta el puerto paralelo del computador con el hardware. Los resultados de la investigaci n permiten concluir que un sistema de reconocimiento de patrones basado en t cnicas de procesamiento de im genes y redes neuronales LVQ clasifica de forma efectiva y robusta dos tipos de botellas.

Palabras claves: sistema, reconocimiento de patrones, procesamiento de im genes, redes neuronales.

ABSTRACT

The purpose of this research was to develop a Visual Pattern Recognition System based on Image Processing and Neural Network Techniques. The investigation is oriented to design and build the system, considered by this, as a projective research, in the same way, according to its method is descriptive, and by its source is documental. The methodology belongs to Cho (2002) designed on the bases established by Angulo (1996). The software is written in Matlab 6.5, and its algorithm uses the image processing techniques to cut and filter the photo, and then processed by a LVQ neural network in the recognition task. The hardware is made of pieces of wood,



piston based actuators and a web cam as the image capture device. Moreover, an interface based of relays and opt couplers was designed and built to connect the PC's parallel port to the hardware. The research results reveal that a Visual Pattern Recognition System Based on Image Processing and LVQ Neural Network Techniques is capable of classifying two kinds of bottles in a robust and effective way.

Keywords: system, pattern recognition, image processing, neural network.

INTRODUCCIÓN

En la actualidad, la automatización industrial tiende a reemplazar la mano del hombre en la totalidad de sus procesos operativos, con la finalidad de incrementar la productividad y reducir el riesgo de exposición física que representa la participación humana. Con estudios avanzados, se han logrado construir sofisticadas máquinas que pueden sustituir la mano humana, pero aún existe un gran vacío en las actividades que requieran emular los cinco sentidos del hombre, los cuales tienen como finalidad común el reconocimiento de patrones. Particularmente útil es la capacidad visual del hombre, para actividades como reconocimiento, clasificación, verificación entre otros. Muchos han sido los estudios que buscan soluciones aplicables en el ámbito industrial que puedan proveer la capacidad visual a las máquinas, pero la complejidad matemática y las exigencias en capacidad de procesamiento, han obstaculizado la construcción de modelos factibles.

Como propuesta a la solución del problema indicado, la presente investigación presenta el diseño y la construcción de un sistema de reconocimiento de patrones, basado en las redes neuronales del tipo LVQ, y en técnicas de procesamiento de imágenes, desarrolladas en el entorno de programación Matlab 6.5.

El procesamiento de imágenes al nivel de bits es el medio para reducir y filtrar la imagen, permitiendo seleccionar únicamente las informaciones útiles de cada imagen, y realizar una pequeña reconstrucción previa al proceso de reconocimiento. La red neuronal LVQ es la técnica matemática utilizada, la cual reduce de forma considerable la complejidad y mejora el tiempo de respuesta, en el proceso de reconocimiento.

El sistema construido es capaz de reconocer y clasificar dos diferentes tipos de botellas, de forma eficiente y robusta, con un tiempo de respuesta aceptable.



OBJETIVO GENERAL

Desarrollar un sistema de reconocimiento de patrones visuales industrial utilizando técnicas de procesamiento de imágenes a través de redes neuronales, para la clasificación de botellas de bebidas.

OBJETIVOS ESPECÍFICOS

- Establecer las diferencias visuales existentes entre las botellas de bebidas seleccionadas. Analizar fundamentos teóricos sobre la visión de máquina y de redes neuronales.
- Diseñar el software de captura y procesamiento de imágenes de botellas de bebidas, basado en teorías de visión de máquinas y redes neuronales.
- Diseñar las partes mecánicas necesarias para la clasificación de botellas.
- Programar el software de reconocimiento de imágenes de acuerdo al diseño previo.
- Construir una estructura mecánica de clasificación de botellas que sirva de apoyo para el proceso de reconocimiento visual. Demostrar la funcionalidad del sistema de reconocimiento de botellas a través de pruebas.

TIPO DE INVESTIGACIÓN

La presente investigación se clasifica por su finalidad como proyectiva, ya que esta orientada a presentar una solución a la necesidad industrial de incluir la capacidad visual a las técnicas usadas para automatizar las operaciones. Según el método, la investigación es de tipo descriptivo, debido a que consiste en la caracterización de un hecho, fenómeno o grupo con el fin de establecer su estructura o comportamiento. A su vez, según la fuente, se considera un estudio documental, puesto que se realizó sobre la base de documentos o revisión bibliográfica.

RESULTADOS DE LA INVESTIGACIÓN

Los resultados de la investigación están dados por el desarrollo de las fases de la metodología Cho (2002), conteniendo el análisis de los datos, los resultados obtenidos y la discusión de los mismos.

FASE 1: DEFINICI N DE LAS ESPECIFICACIONES.

Previo al desarrollo y construcci n del sistema de reconocimiento de patrones visuales, fue necesario establecer las caracter sticas de la variable de entrada y los resultados que se pretende lograr en su salida. Para esto se realiz  el siguiente diagrama:

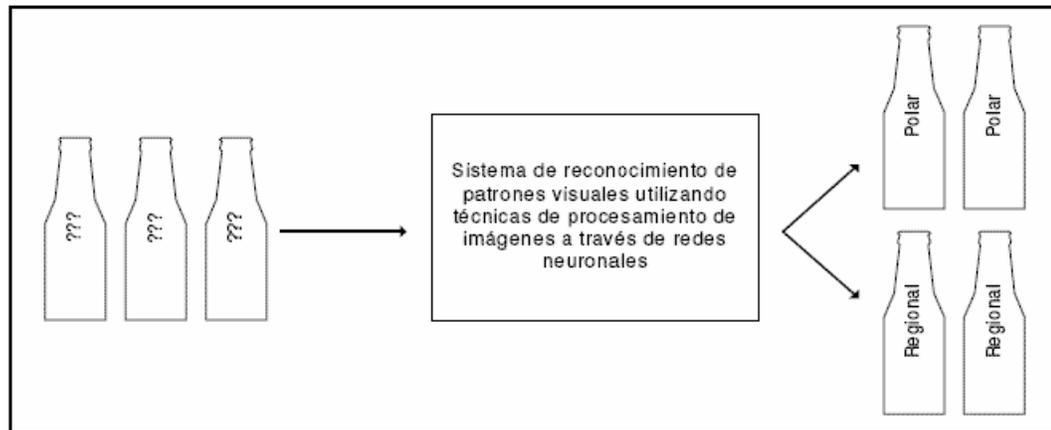


Figura 1. Diagrama de especificaciones del sistema

Fuente: Cho (2002)

Se puede observar que el sistema a desarrollar, tiene como entrada las botellas de identidad desconocida, y como salida las botellas debidamente clasificadas. Se van a utilizar dos tipos de botellas, la de cerveza polar y la otra de regional.

FASE 2: ESQUEMA GENERAL

Una vez identificado las necesidades del sistema, se realiz  un diagrama de bloques, en donde se detallan los diferentes m dulos necesarios y las funciones que desempe an cada uno ellos, as  como tambi n se indican el flujo de las informaciones entre los bloques.

El dise o del diagrama est  basado en las teor as de visi n de m quinas y de redes neuronales recolectadas y presentadas en el cap tulo II de la presente investigaci n. Logrando de esta manera, cumplir el segundo objetivo planteado.

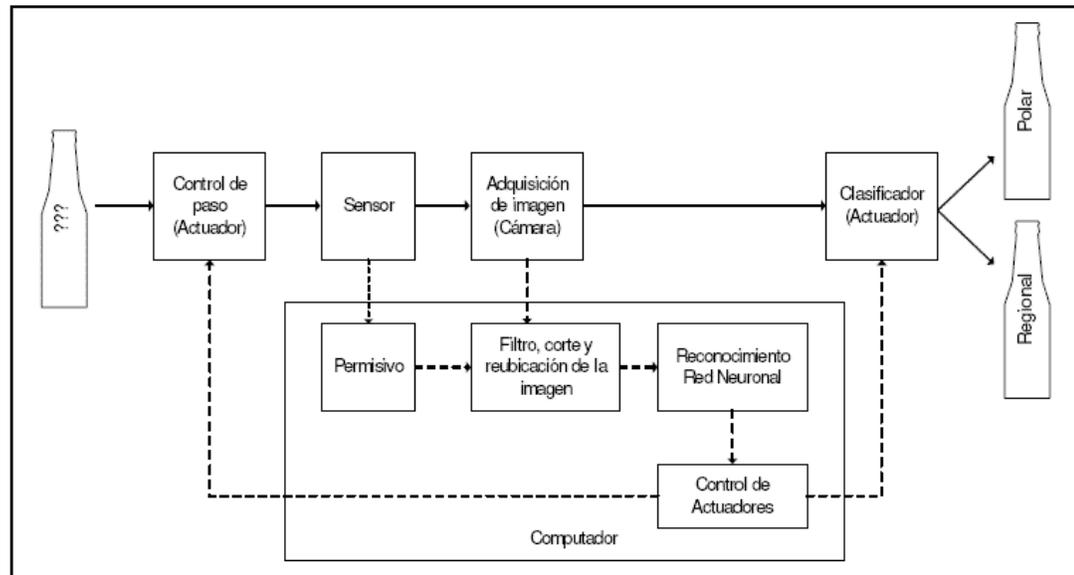


Figura 2. Diagrama de Bloques Funcionales del Sistema.
Fuente: Cho (2002)

A continuaci  n se explican los bloques presentados en el diagrama de la figura 2:

- Botella desconocida: es la entrada del sistema y est   representado por una foto de 160x120 p  xeles 256 colores.
- Control de paso (actuador): dispositivo f  sico que limitar   el paso de la botella.
- Sensor: indicar   la presencia de botella en la zona de adquisici  n de imagen.
- Adquisici  n de imagen (c  mara): dispositivo de captura de imagen.
- Permisivo: bandera activado por sensor e indica presencia de botella.
- Filtro, corte y reubicaci  n de la imagen: conjunto de instrucciones ejecutadas a nivel de software para procesar la imagen y facilitar el reconocimiento.
- Reconocimiento red neuronal: conjunto de funciones matem  ticas dise  adas seg  n la teor  a de la red neuronal, que procesan las im  genes y da como resultado el tipo de botella.

- Control de actuadores: según el resultado de la red neuronal, se activan las señales de control para clasificar las botellas.

- Clasificador (actuadores): dispositivo físico que realiza la clasificación de la botella según lo indique el computador.

- Botellas clasificadas: representa la salida del sistema, las botellas identificadas se representa con un único número de dos posibles valores 1 y 2, para cada uno de los tipos de botella.

Por otra parte, para que el sistema pueda realizar el trabajo de clasificación fue necesario establecer las diferencias visuales que servirán como criterios de identificación. Las botellas presentan diferencias básicamente en su etiquetado, puesto que tanto en forma y color son iguales.

Con la finalidad de facilitar el proceso de reconocimiento y minimizar el costo computacional, se utilizó la etiqueta que se localiza en el cuello como variable para reconocer en el proceso de clasificación. Las razones son su poca similitud visual, reducido área y simetría entre la parte frontal y la posterior. Se observa lo planteado en la figura 3 presentada a continuación:

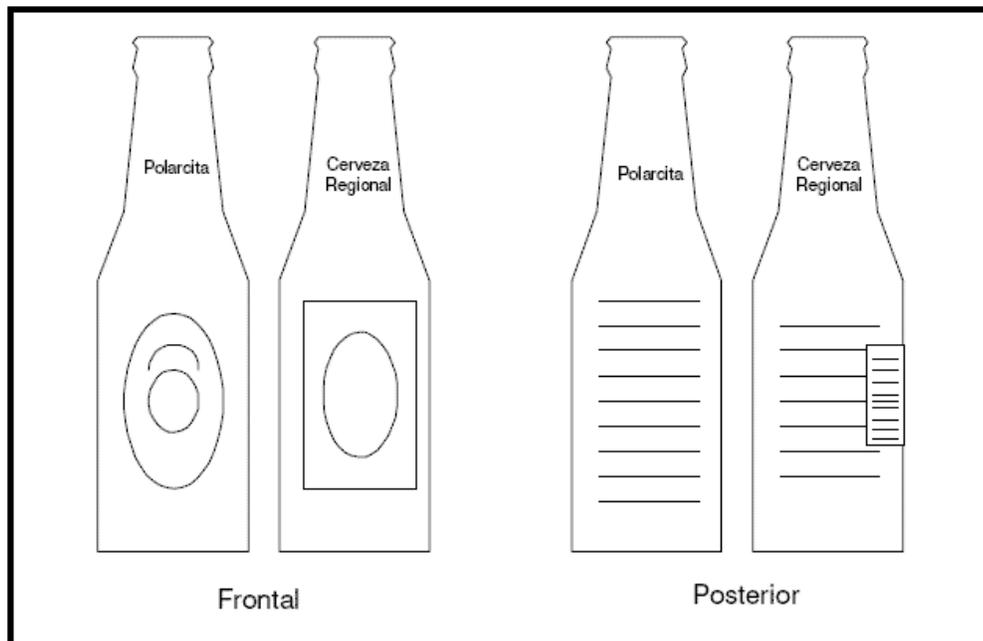


Figura 3. Diferencias visuales entre las botellas
Fuente: Cho (2002)

FASE 3. ORDINOGRAMA GENERAL

Ya definidos los bloques funcionales, se detalla a continuación el diagrama de flujo que regirá el funcionamiento del sistema, en donde se observan las partes que conforman el sistema y su funcionamiento.

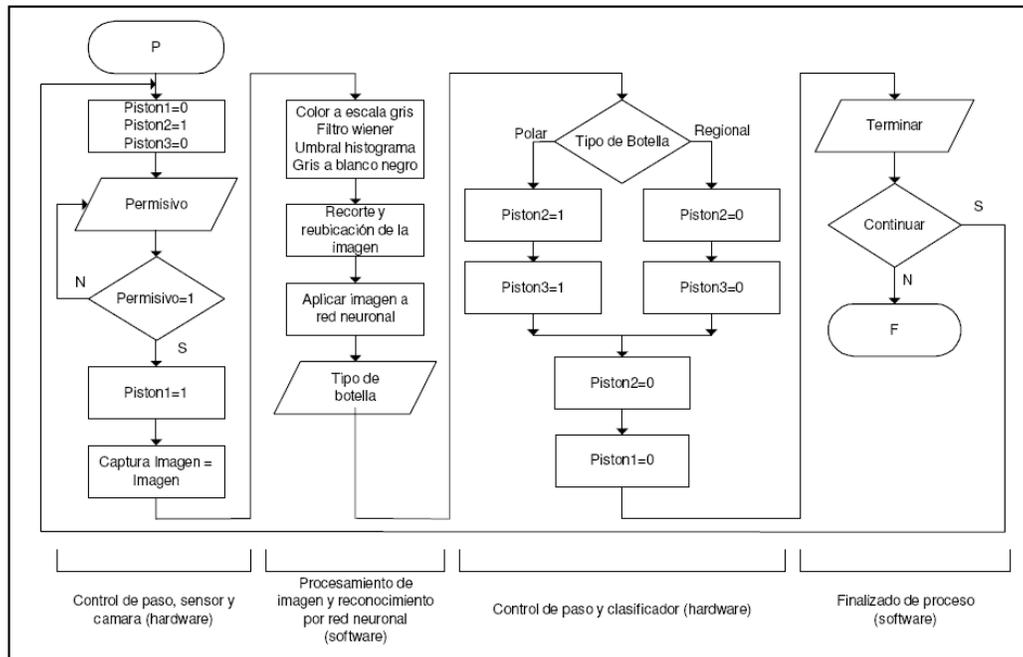


Figura 4. Diagrama de flujo del sistema de reconocimiento

Fuente: Cho (2002)

A continuación, la descripción de los bloques indicados en la figura 4:

- P: Inicio de la rutina
- Pistón 1, Pistón 2, Pistón 3: coloca los pistones en posición inicial.
- Permisivo: verificación de la presencia de botella, de ser positivo, procede con rutina de reconocimiento, de lo contrario, espera.
- Pistón 1: osiciona el pistón 1 para impedir el paso de botellas adicionales
- Captura de imagen: la cámara es activada y toma la foto de la botella.



- Color, filtro, umbral, gris: procesa la imagen aplicando conversión de color a gris, aplica el filtro de ruido, detecta el umbral y convierte la imagen en blanco y negro.
- Recorte y reubicación: limita el tamaño de la imagen a un valor constante, extrae la zona representativa y acomoda la imagen.
- Red neuronal: procesa la imagen con la red previamente entrenada, el resultado será un número escalar que representa el tipo de botella.
- Tipo de botella: salida de la red neuronal que determina la acción de los pistones clasificadores según el tipo de botella.
 - Pistón 2: libera la botella de la zona de captura de imagen.
 - Pistón 3: mueve el dispositivo que separa las botellas según su tipo.
 - Pistón 2: vuelve el pistón 2 a su posición inicial.
 - Pistón 1: libera la siguiente botella.
 - Terminar, Continuar: finaliza la rutina de reconocimiento y vuelve a P.

FASE 4. DEFINICIÓN DEL HARDWARE, EL SOFTWARE Y SU INTERFAZ

A continuación se detallan los criterios establecidos por el investigador en el diseño del hardware, para obtener la mejor relación entre desempeño y costo, como se explica a continuación:

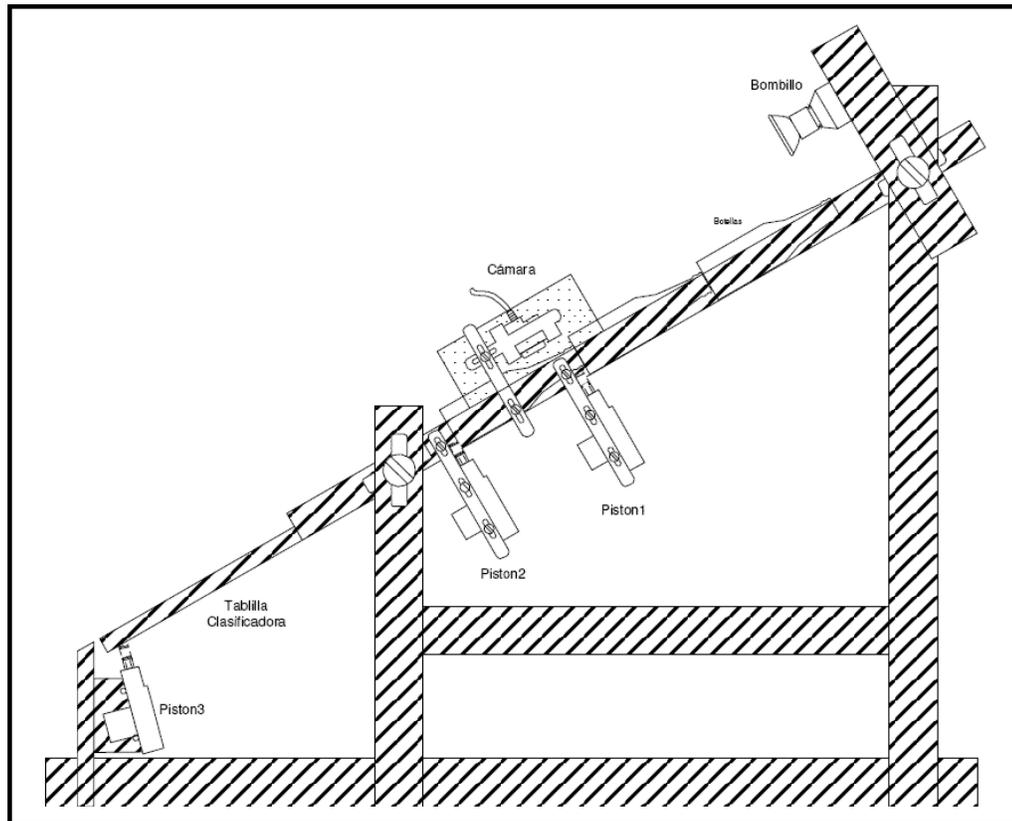


Figura 5. Hardware del sistema de reconocimiento

Fuente: Cho (2002)

El hardware consta de las partes mecánicas del sistema, en ella se realiza las manipulaciones físicas del proceso de clasificación de botellas, desde la introducción de las botellas, su desplazamiento y clasificación. Las principales características del diseño son: capacidad de colocar las botellas en una misma posición en cuando a desplazamiento se refiere, independencia de la iluminación externa, clasificación con solo tres elementos de control y bajo costo.

La línea de desplazamiento consiste en una rampa inclinada en forma de V, el cual aprovecha la energía potencial que proporciona la gravedad, evitando costos en cintas transportadoras. Por otra parte la forma V, permite fijar la botella para la captura de la imagen.

Para la captura de imagen se utiliza una cámara instalada en la rampa por medio de una estructura que cubre la región de captura de imagen para bloquear las luces que provienen en otras direcciones, permitiendo solo



acceso en su parte superior en donde se encuentra un bombillo reflector que ilumina la zona de captura.

El desplazamiento y la clasificación son controlados por tres actuadores en forma de pistón, dos de los cuales retienen las botellas en posiciones específicas, y el tercero, mueve la compuerta de clasificación. En cuanto al software, el sistema está desarrollado en su totalidad en el ambiente matemático Matlab, versión 6.5, el cual integra la computación matemática con funciones de visualización y un lenguaje de gran eficacia ofreciendo un entorno flexible para la realización de computación técnica. Matlab incluye una gran variedad de “toolbox”, de los cuales los utilizados por esta investigación fueron: “image processing (ver. 3.2), neural network (ver. 4.0.2), y data acquisition (ver. 2.2)”.

Estos “toolbox”, con gran cantidad de funciones incorporadas, facilitaron el desarrollo del programa, redujeron el tiempo de experimentación y evitaron la necesidad de interconectar códigos entre diferentes lenguajes de programación.

Finalmente, es necesario establecer la comunicación entre el hardware y el software, y para esto se utilizó el puerto paralelo del computador como puerto de adquisición y salida de datos. Adicionalmente, un montaje electrónico es necesario para realizar las conversiones de voltajes y adaptar las señales del computador a los actuadores y sensores. En las siguientes fases se detallan las especificaciones del circuito electrónico.

Las fases 3 y 4 presentadas anteriormente, satisface los objetivos tres y cuatro de la investigación.

FASE 5: ORDINOGRAMAS MODULARES Y CODIFICACIÓN DE PROGRAMAS

En esta fase se procedió a estructurar el programa, combinando las funciones preconstruidas con rutinas escritas por el investigador. A continuación se presentan las secuencias de las funciones y el trabajo que desempeñan. El listado de los códigos fuentes se encuentran en la sección de anexos.

La parte inicial del programa lo conforman las instrucciones de preparación del puerto paralelo, colocación de los pistones en posición de arranque, y monitoreo constante de la presencia de botella.

La figura 6 muestra el diagrama de flujo de lo mencionado:

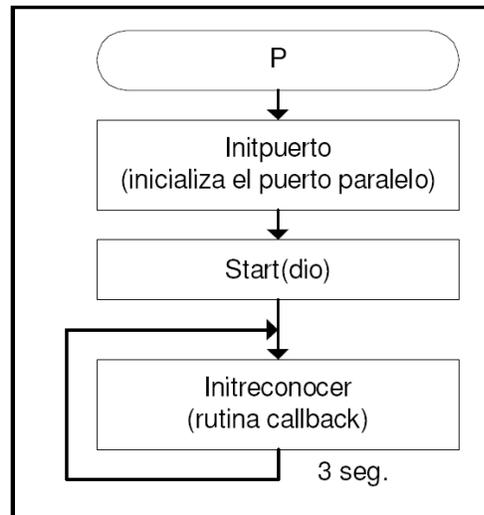


Figura 6. Diagrama de flujo inicial.

Fuente: Cho (2002)

En donde:

P: Inicio del programa.

Initpuerto: utiliza el “toolbox” de adquisici n de datos de Matlab para acceder al puerto paralelo de la computadora, por medio las instrucciones digitalio y addline. La rutina configura el puerto separ ndolo en dos grupos, una entrada y la otra de salida. La primera est  representada por la variable hline1 y consta de 8bit de los cuales se utilizan un par de bit para cada pist n, y la segunda utiliza hline2, formada por 5 bit y s o se usa el primero para tomar la se al del sensor de presencia de botella. Esta instrucci n inicializa los pistones prepar ndolos para la recepci n de botellas. Adem s fija los par metros para ejecutar Initreconocer cada 3 segundos.

Start(dio): activa la ejecuci n de Initreconocer cada 3 segundos.

Initreconocer: activa el comando getvalue para verificar el estado del sensor de presencia de botella almacenando su valor en la variable bot, de ser verdadera, inicia el proceso de reconocimiento ejecutando reconocer.m, y bloquea el paso de nuevas botellas. Luego de reconocer.m, activa los pistones para clasificar la botella.

En la siguiente figura, se puede observar el diagrama de bloque de initreconocer.

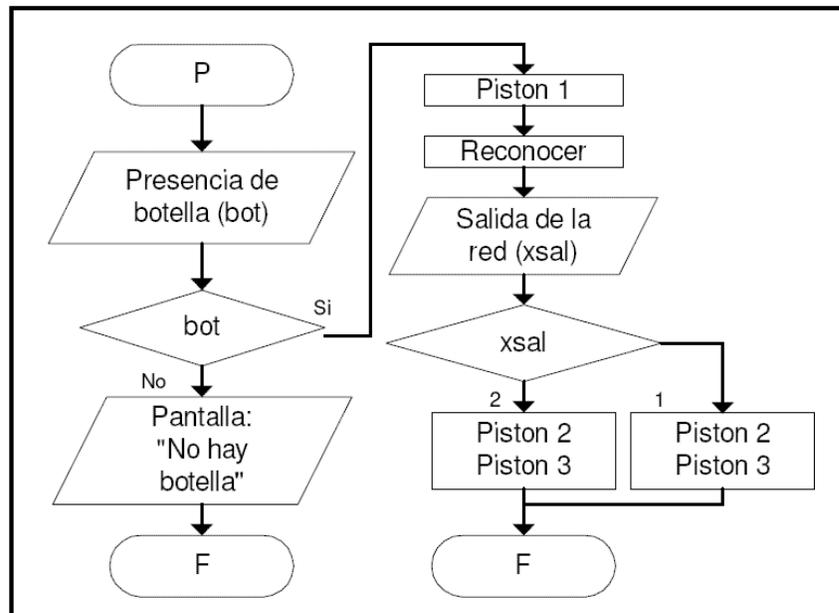


Figura 7. Diagrama de bloque de initreconocer.
Fuente: Cho (2002)

En donde:

- P: inicio de la rutina.
- Presencia de botella (bot): verifica la presencia de botella.
- Bot: bandera que indica el estado de bot, en caso de tener botella, ejecuta reconocer, de lo contrario imprime "no hay botella" .
- Pantalla: muestra la información en el monitor.
- Piston1: cierra el paso a las botellas.
- Reconocer: ejecuta la subrutina de reconocimiento reconocer.m
- Salida de la red: guarda la salida de reconocer.m en la variable sal.
- Xsal: bandera que indica el tipo de botella.
- Pistón 2, pistón 3: activa los actuadores para clasificar las botellas según lo indique xsal.

- F: finalización de la rutina.

El programa principal de reconocimiento, llamado reconocer.m, ejecuta una serie de subrutinas que cumplen funciones específicas. En la figura 8 se observa el diagrama de flujo se observa el ordinograma de reconocer.m.

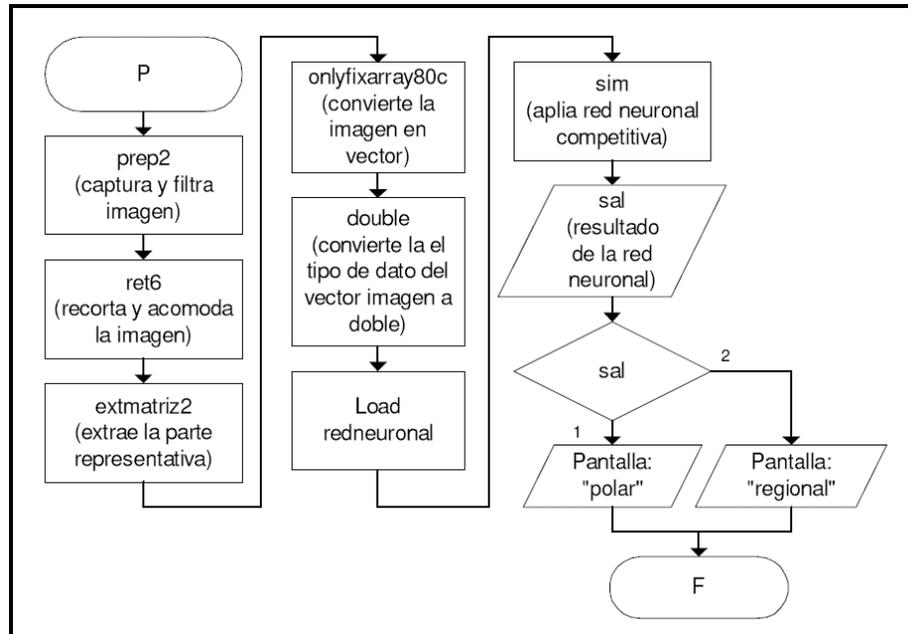


Figura 8. Ordinograma de reconocer.m

Fuente: Cho (2002)

En donde:

- P: inicio de la rutina.
- Prep2: es la primera instrucción del ciclo de reconocimiento y contiene las siguientes instrucciones:
 - vfm: es una función escrita por programadores ajenos a la firma mathworks (propietaria de Matlab), y tiene como función acceder al módulo video for Windows, para tomar control de las cámaras conectadas al computador. La imagen capturada tiene una resolución de 160 x 120 píxeles RGB, es decir cada píxel puede tomar 65536 valores diferentes.
 - Rgb2gray: convierte la imagen de color RGB a escala de grises (256 niveles de gris)



- Wiener2: es un filtro para reducir ruidos en la imagen. Es una funci n que incorpora Matlab 6.5, y fue seleccionada despu s de verificar los efectos visualmente. Se desconoce la f rmula matem tica que la genera.

- Graythresh: devuelve el valor calculado  ptimo del umbral para aplicar la conversi n a blanco y negro. Para el caso de la investigaci n, fue 0.46275 normalizado.

- Img2bw: convierte la imagen filtrada a blanco y negro aplicando el umbral aplicado anteriormente.

- Ret6: reconstruir la imagen original. Ejecuta el enmarcado y recorte de la imagen para tratar de asemejarla a una imagen 0  de resoluci n 45x130 p xeles. En primer lugar se determina la cantidad de secciones que tiene la imagen, si tiene una, desplaza la imagen hacia la derecha o a la izquierda dependiendo de la identificaci n de la secci n presente.

De ser dos, los recorta e identifica las partes, para reubicarlas.

- Extmatriz2: extrae las primeras y las  ltimas cuarenta columnas de la imagen y los une para formar una matriz de 45x80 elementos.

- Onlyfixarray80c: toma la matriz y lo transforma en vector, por medio de instrucciones que coloca cada una de los vectores columnas en una sola gran columna de forma secuencial de 3600x1 elementos.

- Double: transforma el vector imagen de variable enteras de 8 bits a variables de doble precisi n para que pueda ser procesado por la red neuronal.

- Load redneural: carga en memoria la red neuronal competitiva, la cual se utilizar  posteriormente para la identificaci n de la botella. El entrenamiento de la red se realiza en la fase 8, por lo tanto ser  necesario culminar las fases antes de poder determinar los pesos y biases de la red, y la efectividad de la misma.

- Sim: aplica el modelo entrenado de la red neuronal, representado por net, a la vector imagen, para su identificaci n. La salida (sal) es un vector de dos elementos que indica el resultado.

- Sal: es la variable de salida que determina si la imagen aplicada pertenece al grupo polar o al grupo regional.

- Pantalla: muestra el mensaje “polar” o “regional”, seg n indique la variable sal.

- F: fin de la rutina

Con la culminaci n de la codificaci n del programa, seg n dise o previo planteado en la fase 3, se cumple con el quinto objetivo espec fico de la investigaci n.

FASE 6. IMPLEMENTACI N DEL HARDWARE

El hardware fue construido utilizando como material principal las piezas de madera, los cuales fueron trabajadas por un carpintero para seg n el dise o presentado en las fases previas.

Para la iluminaci n se utiliz  un bombillo dicroico de 50W rosca E27 con par bola. Esto permite obtener un haz de luz mucho m s concentrado que los bombillos incandescentes, para asegurar la iluminaci n constante. La c mara de video utilizada es del modelo Logitech QuickCam Web usb. Es de importancia se alar que el sistema requiere la captura de im genes de 160x120 p xeles RGB.

En cuanto a los actuadores, se emplearon solenoides utilizados en los dispositivos de cierre autom tico de veh culos automotores, los cuales son activados con 12 voltios DC y tienen un desplazamiento en pist n de 1,5 cm. En la figura 9 se se alan la forma en que est n conectados la c mara y los pistones:

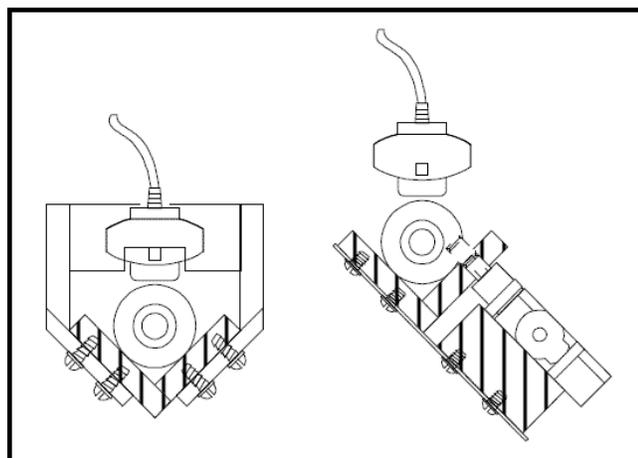


Figura 9. Ubicaci n de la c mara y de los pistones
Fuente: Cho (2002)

El circuito interfaz entre los pistones y la computadora, tienen la función de aislar eléctricamente los dos elementos y el circuito inclusive. Para esto se utilizó optoacopladores y relés de 6v.

Las especificaciones de los materiales utilizados son los siguientes:

Etiqueta	Componente	Modelo	Especificaciones
R1,R2,R3,R4,R5, R6,Ra1 Ra2,Ra3,Ra4,Ra5, Ra6	Resistencia	1K_	1/2W
D1,D2,D3,D4,D4, D6,Da1 Da2,Da3,Da4,Da5, Da6	Diodos	1N4148	Max 70V, 0.2A
T1,T2,T3,T4,T5,T6	Transistor NPN	BC548	BVCBO=75/80V BVCEO=45/80V Icmax=0.6/1A
I1,I2,I3	Optoacopladores	NTE3086	Aislamiento 7500V Max Led = 60mA
L1,L2	Relés	6V	Max 7 ^a
L3,L4,L5,L6	Relés	6V	Max 10 ^a
+6V	Fuente DC 6V	GA-1234	Max 1200mA
+12V	Fuente DC 12V		

Tabla 1. Especificaciones de los materiales electrónicos
Fuente: Cho (2002)

La figura 10 es el diagrama del circuito construido en un protoboard.

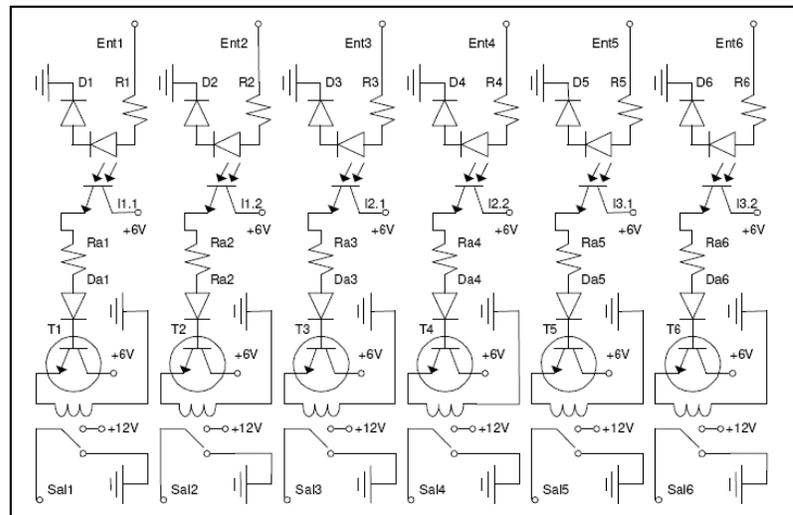


Figura 10. Circuito interfaz.
Fuente: Cho (2002)

La construcci n presentada en esta fase es el resultado que corresponde al sexto objetivo planteado en la investigaci n

FASE 7: DEPURACI N DEL SOFTWARE Y ENTRENAMIENTO RED NEURONAL

Durante esta fase se ejecutaron las rutinas de procesamiento de im genes para comprobar sus funcionamientos, y se realiz  el entrenamiento experimental de la red neuronal para evaluar su desempe o en la identificaci n de la botella.

A continuaci n se muestran una serie de im genes en donde se pueden observar de manera secuencial el efecto que produce cada una de las instrucciones detalladas en la fase 5 de la codificaci n del programa en Matlab.

La imagen capturada por el comando vfm, una foto cuya resoluci n es de 160x120 p xeles RGB se observa en la figura11.

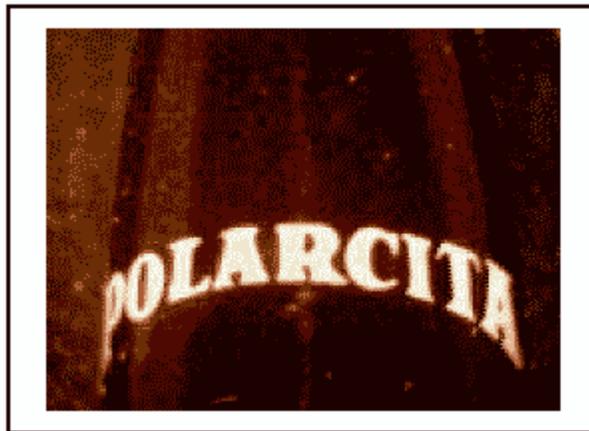


Figura 11. Imagen color botella polar
Fuente: Cho (2002)

Aplicando conversi n a gris, mediante el uso de la funci n rgb2gray, convertimos la imagen de 216 colores a  nicamente 28 escala de grises, como se muestra en la figura 12:

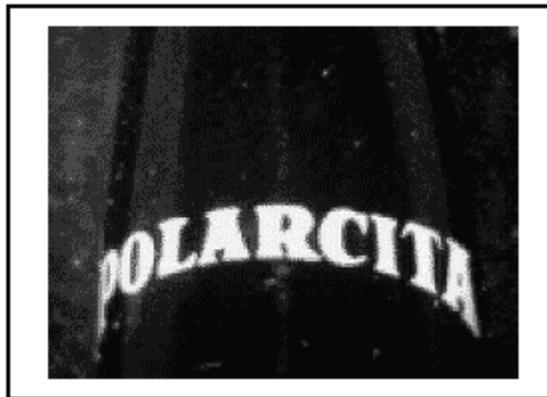


Figura 12. Imagen Polar en escala de grises
Fuente: Cho (2002)

Luego, ejecutando el comand wiener2, incluido dentro del toolbox de procesamiento de im genes de Matlab 6.5 se obtiene lo mostrado en la figura 13:

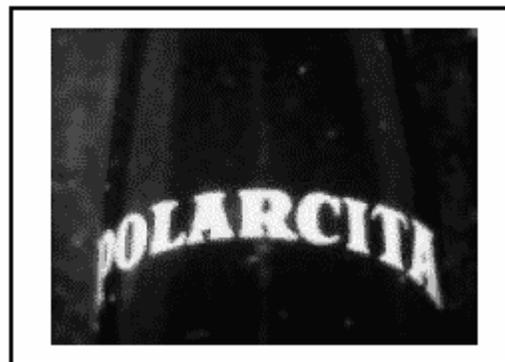


Figura 13. Imagen polar luego de aplicar filtro wiener
Fuente: Cho (2002)

El filtro wiener2 realiza la tarea de suavizar la im genes que fue seleccionado por ensayo y error. Ya con la imagen en escala de grises, se grafic  su histograma como se muestra a continuaci n en la figura 14:

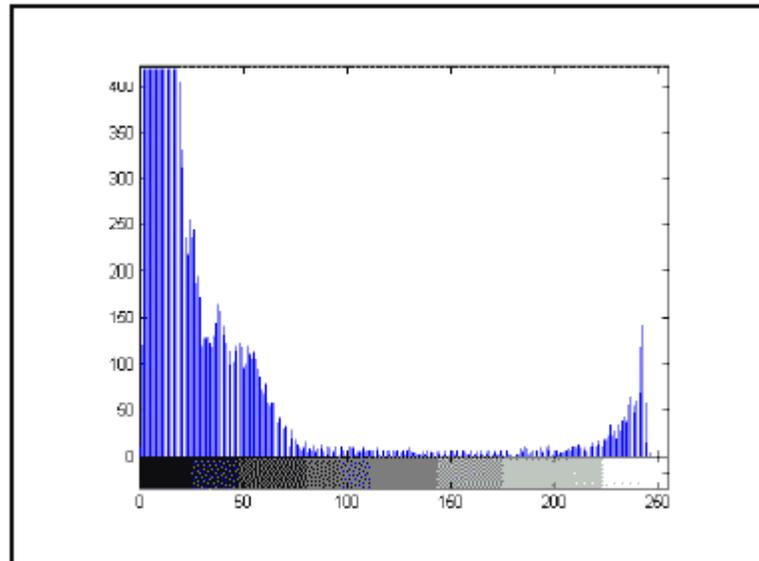


Figura 14. Histograma de la imagen polar.
Fuente: Cho (2002)

Para hallar el umbral de conversi n a blanco y negro, se aplic  el m todo otsu, el valor obtenido de umbral normalizado es de 0.46275. Utilizando este valor se aplic  la conversi n a blanco y negro:



Figura 15. Imagen polar en blanco y negro.
Fuente: Cho (2002)

Ahora se ejecuta la funci n ret6 para de recortar y reconstruir la imagen. En las siguientes figuras se muestran algunas de las posiciones que puede tomar la botella y su respectivo resultado despu s de aplicar ret6.

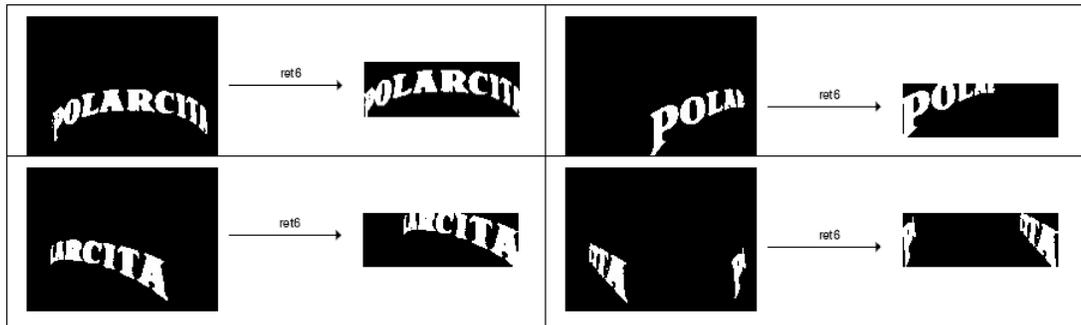


Figura 15: Procesamiento ret6
Fuente: Cho (2002)

Despu s de aplicar la funci n ret6, se extrae las partes representativas utilizando extmatriz2. En la figura 16 se muestra el resultado de en dos instancias.



Figura 16. Im genes procesadas con extmatriz2.
Fuente: Cho (2002)

Con extmatriz2 culminan las manipulaciones de la imagen y se procede con la conversi n de la matriz a vector con la funci n onlyfixarray80c. Debido al gran tama o de las matrices y vectores, de 45x80 y 3600x1 elementos respectivamente, se dificulta ilustrarlo en la presente investigaci n.

Antes de continuar con el proceso de reconocimiento, se realiz  el entrenamiento de la red neuronal, el cual seg n lo indicado en fases anteriores, es del tipo LVQ (Cuantificaci n de vector de aprendizaje).

Para el entrenamiento se capturaron 72 im genes de la botella polar y 72 im genes en diferentes posiciones a posiciones uniformes. El total de muestra fue de 144 im genes que, despu s de aplicarles las funciones de procesamiento, fueron agrupadas en una matriz de 3600x144 elementos.



La red neuronal construída es del tipo LVQ con una primera capa 10 neuronas competitivas y 2 neuronas lineales, entrenadas con el método Kohonen, a una tasa de aprendizaje de 0.05 y 100 epochs.

Previo a la aplicación de la neurona LVQ, se realizaron numerosas pruebas con las neuronas del tipo competitivo, pero los resultados obtenidos no fueron satisfactorios. Las razones por las cuales, la del tipo competitivo falla en realizar una identificación precisa, se debe a la similitud en las imágenes de las botellas de diferentes tipos en algunas posiciones. Se probó incrementando el número de neuronas, pero el resultado no cambió, el entrenamiento no supervisado falla en el proceso de entrenamiento de la red.

Según explicado en el marco teórico, las neuronas LVQ es la alternativa supervisada, para las capas competitivas, puesto que añade un clasificador lineal en la salida de la red. En las primeras pruebas, las neuronas LVQ también fallaron debido que en determinada posición, ambas botellas tienen un área superior al 50% de la imagen en negro. Para solventar esto, se elaboró la instrucción `onlyfixarray80c`, que obvia las 50 columnas del centro de la imagen, reduciendo el área que causa problema de identificación.

Después de realizar el recorte de la imagen, se obtuvieron resultados satisfactorios. Los parámetros asignados, como el número de neuronas en la capa competitiva, fueron obtenidos por ensayo y error.

Los comandos utilizados fueron:

```
net=newlvq(wrank,10,[0.5 0.5])
```

```
net.trainParam.epochs=100
```

```
net.trainParam.lr=0.05
```

```
net=train(net,p,t)
```

En donde, `wrank` es la matriz inicial de los pesos con valores medios de 0.5, `p` es la matriz de entrada constituidos por las imágenes muestras, y una matriz con los resultados conocidos para cada imagen.

En la figura 17 se observa la gráfica del “performance goal” durante los epochs de entrenamiento, se puede interpretar de ella que la red alcanza estabilidad óptima al tercer epochs.

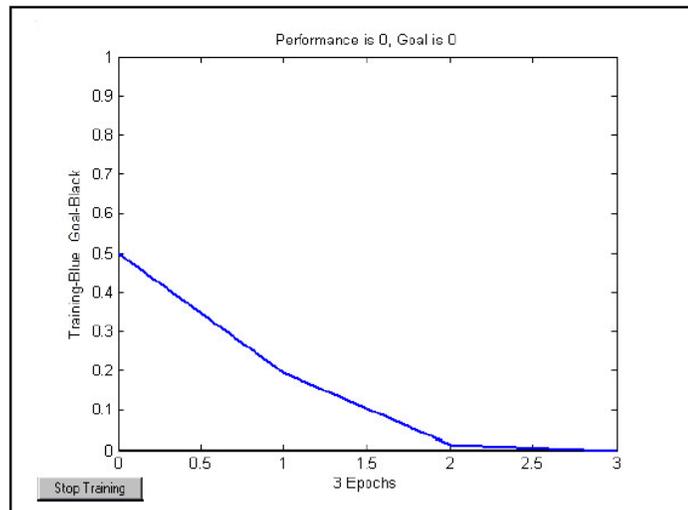


Figura 17. Gr fica del performance vs epoch del entrenamiento.
Fuente: Cho (2002)

Una vez entrenado la red con valores de pesos y biases establecidos como resultado del entrenamiento, se almacenan para ser utilizados posteriormente en el proceso de reconocimiento.

FASE 8: INTEGRACI N DEL HARDWARE Y SOFTWARE.

Ya probados los diferentes m dulos de hardware y software, se efectu  las conexiones finales que para unir las partes obtenidas de las fases previas de la presente investigaci n. Para esto se utiliz  un conector DB25 macho y un cable con sus terminales para conectar la computadora al circuito interfaz, y otros cables del circuito a los actuadores y al sensor.

FASE 9: PRUEBA FINAL.

Para la evaluaci n final del sistema, se escogi  al azar cien botellas de cerveza polar y regional como muestra para probar la capacidad de reconocimiento del sistema.

El resultado obtenido es satisfactorio, puesto que se logr  identificar el 100% de las muestras. A su vez, se cumple con el s ptimo y  ltimo objetivo de la investigaci n.



CONCLUSIONES

Se observó que las botellas de polar y regional de 222ml presentan diferencias a nivel de la impresión, tanto en el cuerpo como en el cuello de la botella. El área a ser sometida al proceso de reconocimiento fue la impresión en el cuello, que es mucho menor en tamaño que la del cuerpo, reduciendo la cantidad de datos a procesar. Se determinó que las técnicas de procesamiento de imágenes de recortar a tamaño de 45x80 píxeles, aplicar el filtro wiener para eliminar ruidos, convertir a blanco y negro con umbral de 0.46275, y la reubicación de la imagen, reconstruye parcialmente la imagen creando un patrón de posición constante, facilitando el proceso de reconocimiento.

Se comprobó que si dos vectores son similares en sus entradas, la capa competitiva, probablemente los colocará dentro de una misma clase.

No existen mecanismos en el diseño de capas competitivas que sea capaz de señalar si dos vectores de entrada son de la misma clase o diferentes.

Se comprobó que la red neuronal LVQ de 10 neuronas en su capa intermedia y un entrenamiento supervisado completado en 3 epoch, realiza el reconocimiento de la imagen en la totalidad de las pruebas realizadas. Una red competitiva de 10 neuronas con entrenamiento no supervisado, no cumple con la actividad. Se observó que el sistema de reconocimiento requiere una luz controlada para anular los efectos de la luz externa, teniendo de esta manera control sobre la apariencia de las imágenes a ser captadas.

Se confirmó que el uso de una rampa tipo V, reduce los desplazamientos de posición de las botellas con respecto a la cámara, a la vez que aprovecha la fuerza de gravedad para clasificar la botella.

Se demostró que el ambiente de programación Matlab 6.5 con los toolboxes de red neuronal, procesamiento de imagen, adquisición de datos y operación de matriz, simplifican y facilitan la construcción del software.

Se constató que un sistema construido en piezas de madera, actuadores tipo pistón y una cámara webcam, funciona para clasificar las botellas, demostrando la aplicabilidad del software diseñado.

Se demostró la aplicabilidad de las técnicas de procesamiento de imágenes y red neuronal en un sistema de reconocimiento de patrones



visuales, por medio de la construcción y diseño de un prototipo que clasifica una botella por cada 3 seg.

REFERENCIAS BIBLIOGRÁFICAS

Angulo, J. (1996). Robótica práctica: tecnología y aplicaciones. España: Editorial Paraninfo

Demuth, H. y Beale, M. (1998). Neural Network for use with Matlab. U.S.A.: The Mathworks, Inc.

Electrónica México, (2001). Redes Neuronales Artificiales. TREC Soluciones. Disponible: <http://www.electronica.com.mx/neural/>, consulta: 2002, Octubre 22.

Erro, L. (1999) Visión Artificial y Robótica. Hemeroteca Virtual ANUIES. Disponible: http://www.hemerodigital.unam.mx/ANUIES/ipn/arte_ciencia_cultura/mar-abr99/paso/sec_4.html, consulta: 2002, Octubre 15.

FU, K.; González, R.; Lee, C. (1998). Robótica: control, detección visión e inteligencia. España: McGraw-Hill.

González, M. (2001). Tema 5: Redes Auto-organizativa. Disponible: <http://www.lfcia.org/~cipenedo/cursos/scx/archivospdf/Tema5-0.pdf>, consulta 2003, Enero 15.

Groover, M.; Weiss, M., Nagel, R.; y Odrey, N. (1994) Robótica Industrial. Mexico: McGraw-Hill.

Haykin, S. (1998) Neural Networks: Foundation. Prentice Hall. Disponible: http://www.infor.uva.es/biometria/documentos/informes_uva/EstadoArte/EstadoArte/node3.html. consulta: 2002, Octubre 21.

Kröse, B. y Smagt, P. (1993) An introduction to neural networks, University of Amsterdam. Disponible: http://www.infor.uva.es/biometria/Documentos/informes_uva/EstadoArte/EstadoArte/node3.html, consulta: 2002, Octubre 21.

Universidad de Chile (2000). Neuronas artificiales y redes neuronales. Universidad de Chile. Disponible: <http://www.ing.udec.cl/~rirraz/memo/basic/neu1.html> consulta: 2002, Octubre 22.