

BUSQUEDA DE PATRONES

Tabla de contenido

Introducción	3
Definición del problema	3
Objetivos	3
Objetivo general	3
Objetivo específico	3
Marco Teórico	3
Modelo o patrón.	4
Búsqueda de patrones	4
Introducción al lenguaje.....	5
El lenguaje	6
Desarrollo	7
El algoritmo	7
El programa	8
Resultados	11
Conclusiones	12
Bibliografía	12

Introducción

Búsqueda de patrones por medio de una cámara web usando Labview el cual busca un patrón determinado dentro de un entorno. El presente proyecto se realizó con la finalidad de investigar la visión artificial para determinar en qué sectores podremos utilizar nuestro proyecto a futuro.

Labview es una herramienta gráfica de test, control y diseño mediante la programación. El lenguaje que usa se llama lenguaje G.

El método usado para nuestra búsqueda de patrones es por medio de vectorización el cual consiste representar las imágenes como si fueran vectores y así facilitar nuestra búsqueda.

Las limitantes de nuestro proyecto es que solo abarca imágenes o video para el reconocimiento de patrones.

Definición del problema

El problema reside en que se debe de encontrar un patrón dado dentro de un entorno dado, el cual será tomado por una cámara web. Al momento de que la cámara encuentre ese patrón, la aplicación deberá de mostrar la ubicación en la que se encuentra y determinar que sea el mismo patrón que se dio en un principio.

Objetivos

Objetivo general

Buscar un patrón dentro de un entorno dado, ya sea por medio de video o por imágenes. Con el fin de interpretar fotografías, reconocer imágenes, reconocer caras, identificar objetos, etc.

Objetivo específico

Encontrar un patrón específico por medio de una cámara web utilizando Labview.

Marco Teórico

El reconocimiento de patrones también llamado lectura de patrones, identificación de figuras y reconocimiento de formas consiste en el reconocimiento de patrones de señales. Los patrones se obtienen a partir de los procesos de segmentación, extracción de características y descripción dónde cada objeto queda representado por una colección de descriptores. El sistema de reconocimiento debe asignar a cada objeto su categoría o clase (conjunto de entidades que comparten alguna característica que las diferencia del resto). Para poder reconocer los patrones se siguen los siguientes procesos:

-Adquisición de datos

-Extracción de características

-Toma de decisiones

El punto esencial del reconocimiento de patrones es la clasificación: se quiere clasificar una señal dependiendo de sus características. Señales, características y clases pueden ser de cualquiera forma, por ejemplo se puede clasificar imágenes digitales de letras en las clases «A» a «Z» dependiendo de sus píxeles o se puede clasificar ruidos de cantos de los pájaros en clases de órdenes aviares dependiendo de las frecuencias.

Modelo o patrón.

Un modelo es una descripción global del conjunto de datos. Toma una perspectiva completa y total. En contraste un patrón es una propiedad local de los datos, tal vez sólo la tienen ciertas instancias o atributos.

Un patrón es la descripción de la región segmentada

El patrón se forma, usualmente, a partir de una combinación de propiedades (x_1, x_2, \dots, x_n), la cual define el espacio de propiedades en la que actúa el clasificador.

El clasificador no reconoce objetos, reconoce el patrón que define al objeto y si se conoce el patrón que define a un objeto cierto, asigna la región al objeto.

Búsqueda de patrones

Un sistema de reconocimiento de patrones completo consiste en:

- Un sensor que recoge las observaciones a clasificar.
- Un sistema de extracción de características transforma la información observada en valores numéricos o simbólicos.
- Un sistema de clasificación o descripción que, basado en las características extraídas, clasifica la medición.

Por ejemplo se pueden clasificar imágenes digitales de letras en las clases «A» a «Z» dependiente de sus píxeles o se pueden clasificar huellas dactilares.

Las características de la imagen pueden ser:

-Topológicas: número de componentes conexas, agujeros,...

-Geométricas: área, perímetro, curvatura,...

-Estadísticas: momentos,...

Un patrón es un conjunto de características.

Una clase de patrones es un conjunto de patrones “similares”.

El objetivo del reconocimiento de patrones es asignar un patrón a la clase a la que pertenece (lo más automáticamente posible).

Introducción al lenguaje

LabVIEW es una herramienta gráfica de test, control y diseño mediante la programación. El lenguaje que usa se llama lenguaje G.

Este programa fue creado por National Instruments (1976) para funcionar sobre máquinas MAC, salió al mercado por primera vez en 1986. Ahora está disponible para las plataformas Windows, UNIX, MAC y Linux y ha llegado a la versión 8.20 y 8.21 con soporte para Windows Vista.

Los programas hechos con LabVIEW se llaman VI (Virtual Instrument), lo que da una idea de su uso en origen, el control de instrumentos. El lema de LabVIEW es: "La potencia está en el Software". Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Test, Control y Diseño) y el permitir la entrada a la informática a programadores no expertos. Esto no significa que la empresa haga únicamente software, sino que busca combinar este software con todo tipo de hardware, tanto propio -tarjetas de adquisición de datos, PAC, Visión, y otro Hardware- como de terceras empresas.

Principales usos:

- Es usado principalmente por ingenieros y científicos para tareas como:
- Adquisición de datos
- Control de instrumentos
- Automatización industrial o PAC (Controlador de Automatización Programable)
- Diseño de control: prototipaje rápido y hardware-en-el-bucle (HIL)
- Diseño Embebido
- Domótica²⁷

El lenguaje G es de alto nivel cuando se utiliza sobre su interfaz gráfica de “drag and drop” (arrastrar y soltar << sistema de bloques que se enlazan entre si o introducen funciones en una pagina de proyecto para diseño y construcción de programas >>) sin embargo se tiene también una interfaz para programar en línea de código de un lenguaje de alto nivel, también permite la introducción de

espacios tanto para lenguajes como C# y C++®, Java®, Cobol® entre los más usados como bloques de activeX sobre máquina virtual.

Se Utilizará los bloques enlazados en su interfaz gráfica, no solo porque no es necesario llegar a un mayor nivel de programación debido a que las funciones de las que se dispone son suficientes, sino porque es extremadamente sencillo programar de esta forma.

El lenguaje

El algoritmo tiene un proceso sencillo el cual fue buscado con el propósito de aligerar los cálculos que tendrá que hacer el procesador o la unidad de procesamiento central.

Si bien estos cálculos son sencillos, se necesita una gran cantidad debido al flujo de información; si se ven estos flujos se observa que cada cálculo que se realice se lo tiene que hacer sobre cada captura de imagen que nos entregue la cámara ya que sería la menor cantidad de errores y el mejor rendimiento que se podría lograr.

Una cámara estándar en operación normal entrega 30 cuadros por segundo en formato NTSC (National Television System Comité) el cual es el más usado en los Estados Unidos, además del PAL (Phase Alternating Line) que predomina en Europa.

Si una cámara nos entrega 30 cuadros por segundo, el tiempo para cada cálculo es el de un treintavo ($1/30$) de segundo o 0.034 segundos, en este tiempo se deben realizar todos los cálculos del algoritmo incluida la transferencia de los datos o el despliegue de los mismos.

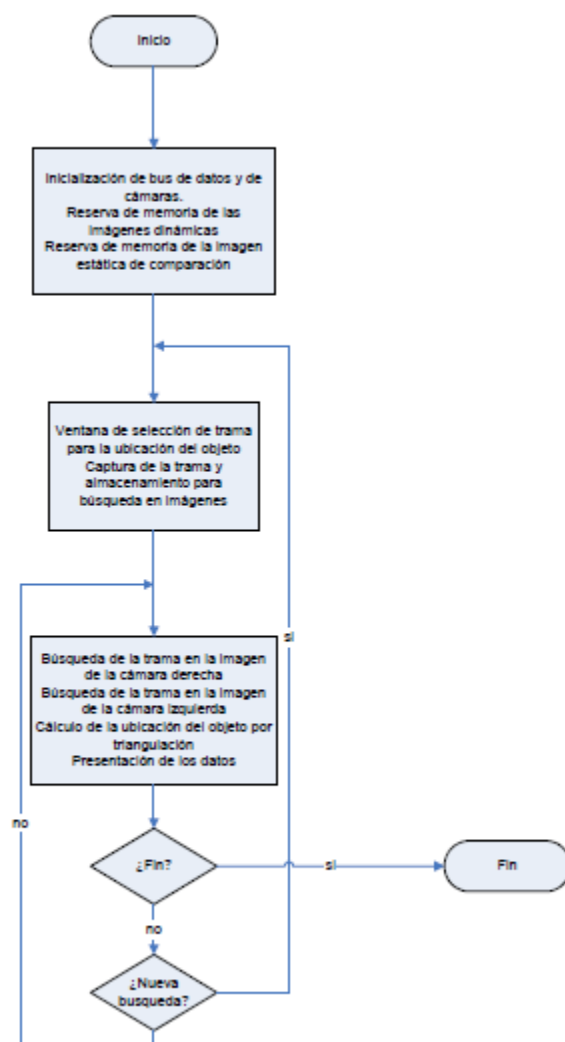
Para el desarrollo del algoritmo se escogerá un lenguaje de programación de “alto nivel” debido a los procesos complicados que se tienen para ubicar una trama dentro de una imagen, un lenguaje de alto nivel no dispone de comandos o acciones para simplificar el proceso de programación.

Los lenguajes de programación de alto nivel no tienen las librerías ni funciones probadas ni optimizadas para estas funciones, si bien estas librerías de funciones pueden ser encontradas para lenguajes como C# o C++, estas son pagadas y de precios elevados; lenguajes matemáticos como Phyton tienen librerías con procesos pesados (cientos de megas para descargar desde el Internet) e igualmente costosas; los lenguajes de programación de medio o bajo nivel quedan descartados por su extrema complejidad para el propósito de desarrollo de este proceso.

Desarrollo

El algoritmo

Un diagrama genérico del algoritmo se presenta en la siguiente figura:



En el primer bloque están concentradas las inicializaciones de los dispositivos y de los bloques de memoria que se reservarán para los cálculos y procesos del algoritmo. En cualquier lenguaje de programación será necesario este proceso.

El segundo bloque está destinado a comenzar el ciclo ininterrumpido de la búsqueda de la trama que necesariamente debe comenzar con la especificación de la misma, luego esta se tiene que almacenar para las futuras búsquedas.

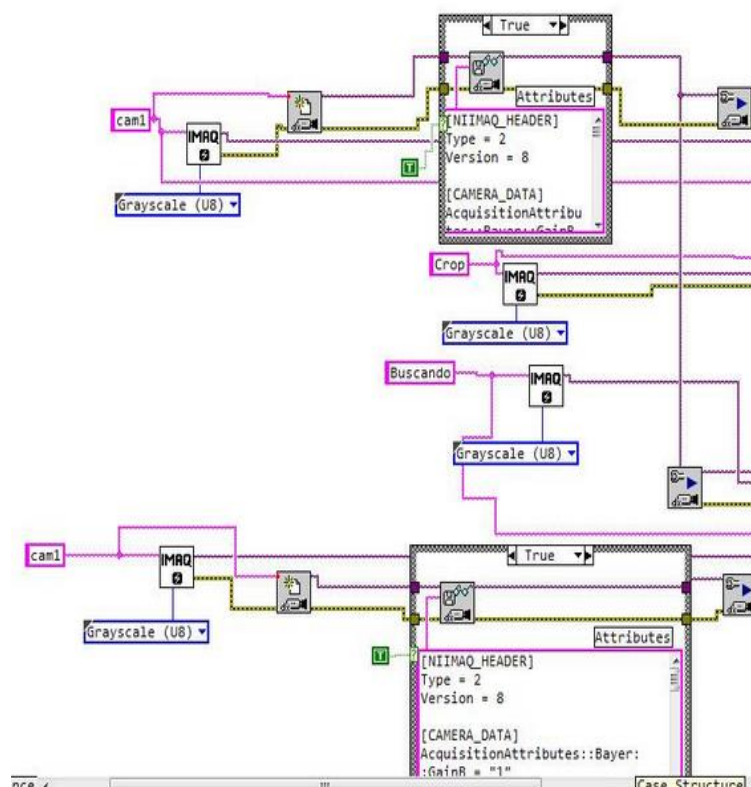
El siguiente bloque es en donde el algoritmo entra en funcionamiento; en él se busca la trama en cada una de las imágenes y con ello se hacen los cálculos para la ubicación del objeto en el espacio.

El bloque de pregunta fin determina si se debe o no salir del ciclo ininterrumpido.

El bloque de pregunta de nueva búsqueda permite salir del ciclo de búsqueda y regresa al bloque de inserción de la trama para la nueva búsqueda que se supone se tomará como un nuevo objeto.

El programa

A continuación explicaremos el programa por medio de bloques para facilitar su comprensión



1.- En este bloque se muestra la reserva de memoria (cam0, crop, buscando, cam1) y configuración de las cámaras firewire en donde se ve que se las configuró en 8 bits de escala de grises; también existe un bloque que inicializa la captura de imágenes de las cámaras y el envío continuo de imágenes hasta que el programa termine.

Con este bloque se realiza dentro del algoritmo mostrado en la figura 5.1 los pasos que inicializan el bus de datos y d cámaras, reservas de memorias dinámicas (cam0 y cam1) y la reserva de la memoria estática para la trama de comparación (crop y buscando).



IMAQ Create.- Es el bloque destinado a las reservas de memoria, los textos cam0, crop, buscando y cam1 son los nombres referenciales de estos espacios de memoria

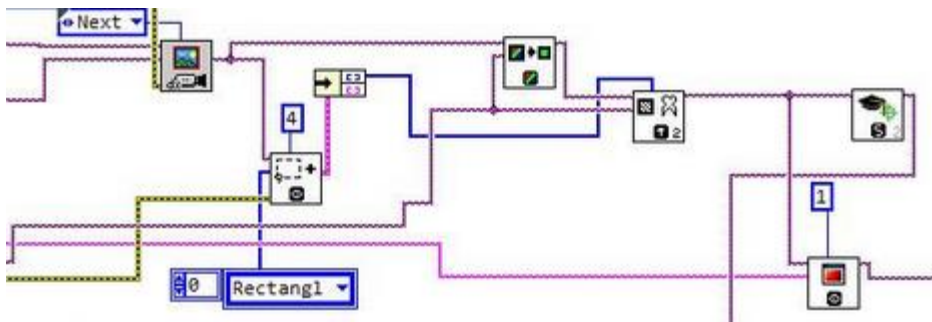


IMAQ1394 Init.- habilita la transferencia de datos con las cámaras IEEE 1394, solicita las capacidades de la cámara, carga el archivo de las configuraciones de las cámaras.



IMAQ1394 Grab Setup.- Configura e inicializa el manejo de adquisición de datos con dispositivos IEEE 1394 ininterrumpidamente.

2.- Aquí están los parámetros necesarios para que se haga la introducción de la trama del objeto a buscar, en este bloque se realizan los pasos de ventana de selección de trama para la ubicación del objeto, captura de la trama y almacenamiento para búsqueda en imágenes, estos dos pasos son los comprendidos en el segundo bloque del algoritmo.



IMAQ ConstructROI.- El bloque de búsqueda de ROI (Region of Interest) que permitirá capturar la imagen de la cámara en ese instante y almacenarla en el espacio de memoria "Crop" además presenta la imagen en una ventana con identificador 1 y también entra en un bloque de aprendizaje de tramas.



IMAQ1394 Get Image.- adquiere el cuadro específico almacenado en el bufer del puerto IEEE 1394 correspondiente a la sesión del bloque IMAQ1394 Grab Setup previamente configurado



IMAQ Extract.- Recorta el espacio deseado de una imagen y lo almacena en un espacio de memoria (crop), con este recorte se piensa hacer las posteriores búsquedas de esta trama

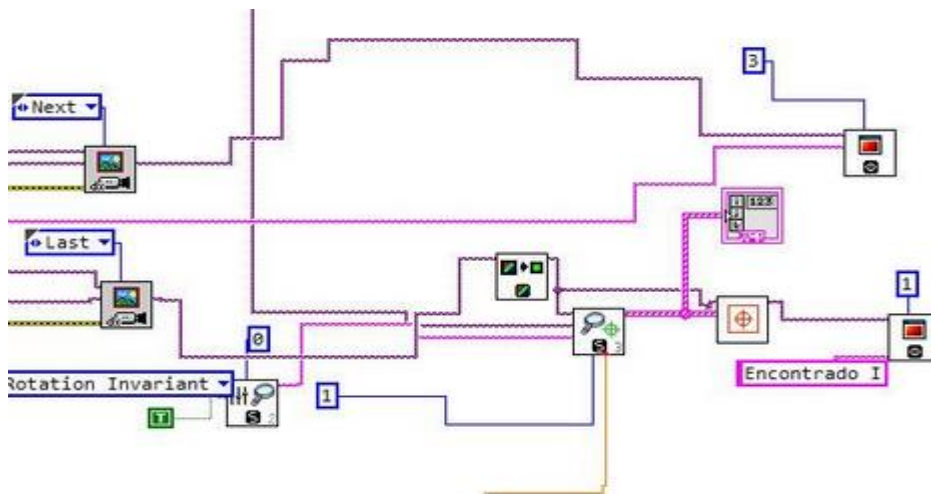


IMAQ Learn Pattern 2.- Crea una descripción de la plantilla con la cual se piensa hacer la comparación en la fase de búsqueda.



El bloque de selección de ROI permite desplegar una ventana para la selección de la región de interés, esta ventana está con el subíndice 4 y se preseleccionó la selección de rectángulo; esto es necesario debido a que el corte de la imagen se realizó con un recortador de imágenes por coordenadas, el cual acepta dos puntos y con ellos realiza el corte rectangular de la imagen.

3.- Este bloque es el encargado de buscar la trama en las imágenes. En la primera parte están las capturas de la trama, luego el bloque que se encarga de programar la forma en la que la trama se buscará, en él está preseleccionado la rotación invariante lo que permite buscar la trama incluso si esta está girada; la calificación de la trama está en 500 en donde 1000 es una copia exacta de la trama y 0 es una trama completamente diferente a la buscada; la búsqueda se la realiza en las dos imágenes capturadas, tanto de la cámara derecha como la de la izquierda.



IMAQ Setup Match Pattern -Fija los parámetros para la fase de búsqueda, en el programa se seteó la rotación invariante con el objeto de hacer las búsquedas incluso si la trama esta en posición oblicua y la búsqueda en subpixeles para permitir pequeñas deformaciones de la trama provocadas por el paralaje.



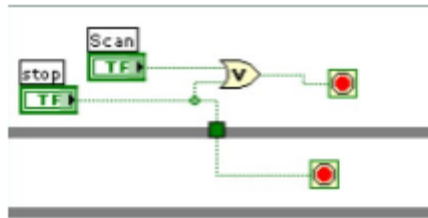
IMAQ Match Pattern - Realiza la búsqueda de una trama en una imagen a inspeccionar. Para este programa se calibraron los resultados en al menos 500 puntos, y se restringió a un solo resultado en la búsqueda, este bloque realiza una asignación de puntos con respecto a la diferencia

que existe entre el histograma de la trama a buscar y la imagen a inspeccionar, de todos los posibles resultados el mejor se presenta debido a que se restringió a un solo resultado.



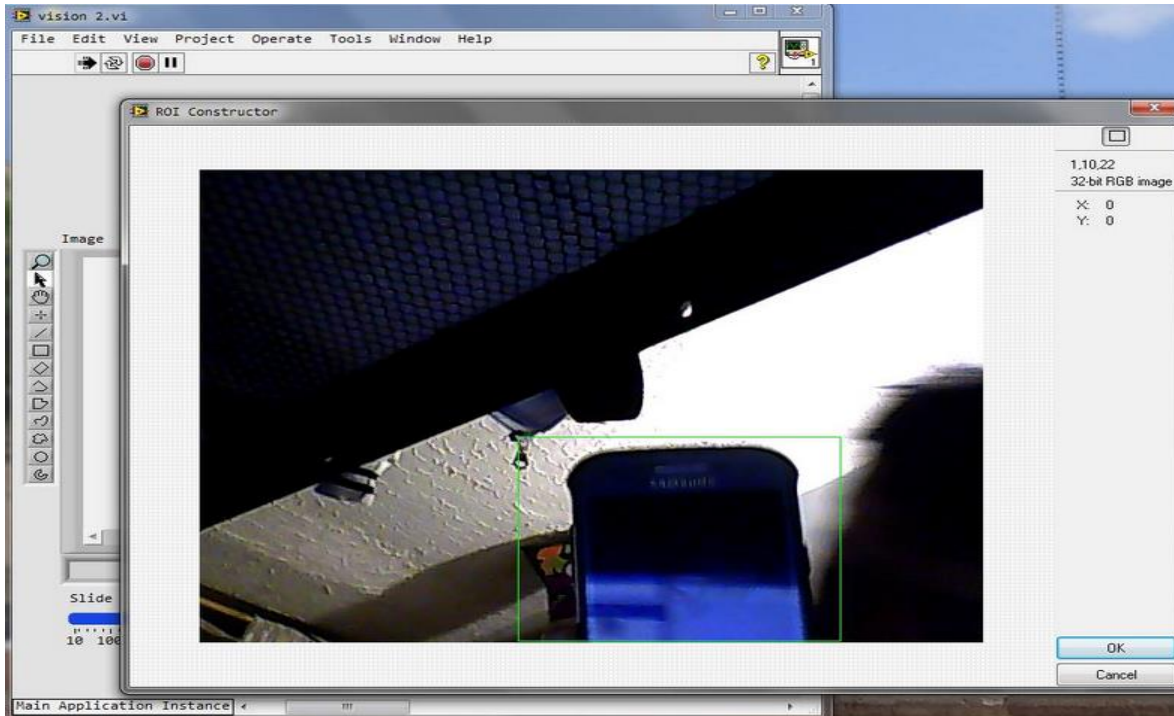
IMAQ WindDraw VI-Muestra la imagen en una ventana exterior

4.- Aquí están los dos bloques de salida del programa: el primero permite salir del bloque de bucle de búsqueda para entrar al bloque de inserción de la trama a buscar, el otro es el botón que permite salir del programa en general.

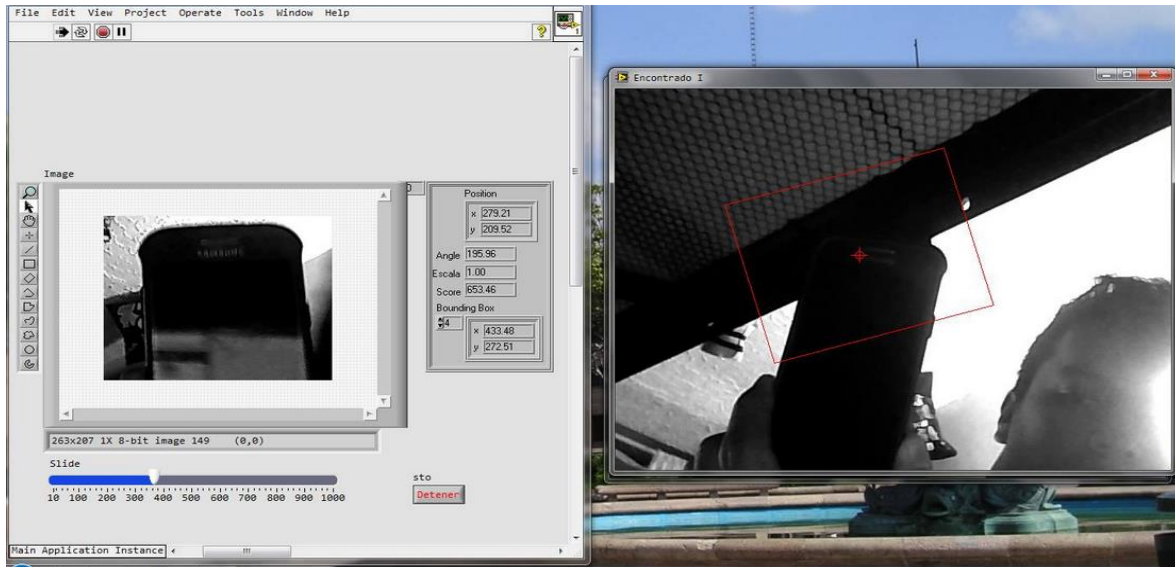


Resultados

Primero por medio de nuestra cámara web tomaremos una foto en la cual estará nuestro patrón a buscar y lo recortaremos con una selección rectangular.



Una vez recortado nuestro patrón a buscar este lo mostrará en la ventana del lado izquierdo y en nuestra ventana derecha empezara buscar nuestro patrón y lo resaltara de forma que podamos visualizar que a encontrado el patrón, también podremos observar la posición en la que se encuentra así como el ángulo.



Conclusiones

La elaboración proyecto arrojó los resultados esperados ya que si encuentra nuestros patrones a buscar y cumplió con nuestros objetivos propuestos, también al momento de realizar el proyecto nos dimos cuenta que si se la da un seguimiento a futuro podremos llegar a identificar rostros, huellas digitales o patrones ya mas específicos que conlleven un mayor nivel de dificultad, así como implantaciones en la industria que produzcan materiales de forma masiva para la detección de imperfecciones en sus materiales de producción.

Bibliografía

- [1] *Visión por Computador* Luis Baumela <http://www.dia.fi.upm.es/~lbaumela/Alcala> Departamento de Inteligencia Artificial Universidad Politécnica de Madrid
- [2] <http://es.wikipedia.org/wiki/Biónica> consultado al 30 de noviembre de 2014
- [3] http://es.wikipedia.org/wiki/Visión_artificial Visión artificial consultado al 30 de noviembre de 2014
- [4] *Image processing with LabView and IMAQ Vision* Thomas Kilinger Prentice Hall PTR