



**CIDESI**

**Centro de Ingeniería y Desarrollo Industrial**

005393

**Proyecto Industrial Terminal**

**Creación de librerías para el procesamiento de imágenes utilizando LabVIEW**

**PARA OBTENER LA ESPECIALIDAD EN  
“TECNÓLOGO EN MECATRÓNICA”**

PRESENTA

Alumno: López López Einer Loreto

Tutor de Planta: Dr. Jesús Carlos Pedraza Ortega

Tutor Académico: Dr. Luis Govinda García Valdovinos

QUERETARO, QRO. Septiembre 2007.



## **INDICE:**

TITULO	2
RESUMEN:	3
ANTECEDENTES:	4
DEFINICIÓN DEL TEMA	5
JUSTIFICACIÓN	6
OBJETIVOS	7
FUNDAMENTOS	8
ELEMENTOS DE LOS SISTEMAS DE PROCESAMIENTO DE IMÁGENES	8
VI'S DE ADQUISICIÓN.	10
BUFFER SIMPLE DE ADQUISICIÓN MEDIANTE VI'S	11
PROCEDIMIENTO	13
INICIALIZACIÓN DE UNA WEBCAM E INTEGRACIÓN DE SUS LIBRERÍAS EN LABVIEW	13
CREACIÓN DE SUB-VI PALETTE TYPE.VI	20
CREACIÓN DE SUB-VI, CONVERSION A 8 BITS Y NEGATIVO	24
CONCLUSIONES	28
BIBLIOGRAFIA	29

**TITULO:**

Creación de librerías para el procesamiento de imágenes utilizando LabVIEW

## **RESUMEN:**

Este proyecto consiste en la creación de librerías de procesamiento de imágenes, con el fin de poder utilizarlas como herramienta de desarrollo preliminar en futuras aplicaciones donde se requiera del uso de sistemas de visión, así como también dar una descripción detallada de cual es el uso y la aplicación de cada librería.

Utilizando herramientas de software de National Instruments, como lo son Vision Assistant 7.1 y LabVIEW 7.1, junto con un sistema sencillo de adquisición de imágenes (webcam), se puede tener un sistema de bajo costo y múltiple funcionalidad que pueda servir como primera referencia en prácticas y proyectos que requieran utilizar un sensor de visión.

## **ANTECEDENTES:**

El interés en los métodos de procesamiento digital de imágenes emergen de dos áreas principales: mejoramiento de la información visual para la interpretación humana, y en la percepción de máquinas para la interpretación automática de datos.

El uso de técnicas computacionales para la mejora de imágenes se llevó a cabo por primera vez en el año de 1964 cuando el Ranger 7 transmitió imágenes de la luna y era necesario corregir varios tipos de distorsiones de las imágenes. Desde ese entonces, el procesamiento de imágenes digitales comenzó a desarrollarse vertiginosamente para solucionar diversos tipos de problemas.

En la industria, las aplicaciones del procesamiento de imágenes para la percepción visual automática abarcan desde el reconocimiento de objetos, reconocimiento de posición, chequeo de forma y dimensión, inspección de superficies y reconstrucción.

## **DEFINICIÓN DEL TEMA**

El presente trabajo se centra en la propuesta de creación e implementación de rutinas de instrumentación virtual basadas en el software LabVIEW, así como del uso de librerías de Vision Assistant, las cuales a través de una herramienta grafica se puede generar un instrumento virtual. La finalidad de uso de LabVIEW y Vision Assistant es a partir de la creación de instrumentos virtuales sencillos se pueda generar una librería de funciones que faciliten el desarrollo de aplicaciones mas complejas en donde se requiera aplicaciones de visión por computadora.

Un sistema de visión consta esencialmente de 4 partes; adquisición, almacenamiento, procesamiento y visualización. La propuesta en este trabajo es el de sustituir sistemas de adquisición y almacenamiento costosos por el uso de una web cam genérica, lo que reduce tiempo y costo en una implementación inicial en algún proyecto que requiera este tipo de sensores. Además el desarrollo de librerías más sencillas de manejar, sin tomar en cuenta todo el diagrama de bloques de instrumentación que la contienen, solamente se visualizaran las entradas y salidas necesarias para la aplicación de las mismas.

Por ultimo, la visualización se llevara a cabo en la misma computadora, la cual puede ser una computadora personal, sin la necesidad de tener algún monitor extra en el sistema de visión.

## **JUSTIFICACIÓN**

Mediante este proyecto se busca simplificar el trabajo al realizar proyectos de procesamiento de imágenes, mediante el uso de las librerías que se utilizan, de forma más constante y cotidiana, así mismo busca realizar un procesamiento de bajo costo, al ser implementada una cámara Web como el dispositivo que sirve para determinar presencia, ubicar posición, medir, e identificar un objeto.

## **OBJETIVOS**

Conocer a detalle los dispositivos de adquisición de imágenes.

Crear librerías mediante LabVIEW, para el procesamiento de imágenes.



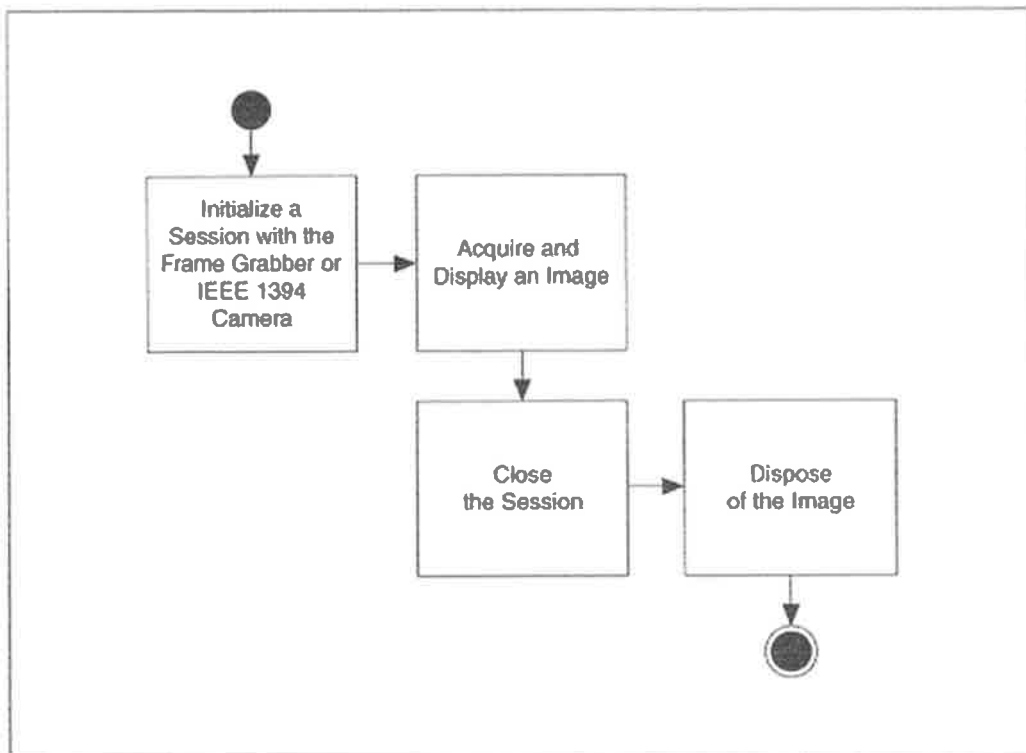
## FUNDAMENTOS

### 1.- Elementos de los sistemas de procesamiento de imágenes

Un sistema de procesamiento digital de imágenes de propósito general consiste de varios elementos o dispositivos para llevar a cabo las operaciones. Estos elementos o dispositivos generalmente llevan a cabo los siguientes trabajos.

- Adquisición de la imagen
- Almacenamiento de la imagen
- Procesamiento de la imagen
- Despliegue de la imagen

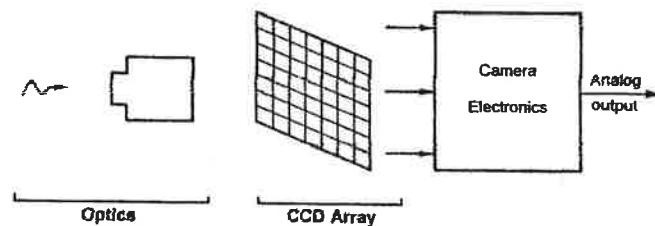
El siguiente diagrama de bloques demuestra el proceso de la adquisición de imágenes:



*Figura 1.- Diagrama de Bloques de adquisición de imágenes*

*Elementos para la adquisición de la imagen:* Los elementos principales para la adquisición de imágenes consisten, primeramente, en el dispositivo sensor, capaz de detectar la energía luminosa y producir una señal eléctrica proporcional a la energía de luz detectada. En segundo lugar, está el dispositivo digitalizador o convertidor análogo-digital que sirve para convertir la señal analógica generada por el sensor de energía luminosa, a una señal digital.

El dispositivo más comúnmente usado para registrar imágenes electrónicamente es el CCD (Charge-Coupled Devices), que es el corazón de una cámara electrónica de estado sólido. Estos dispositivos proveen la conversión de intensidad de luz a señales eléctricas. La Figura muestra un diagrama esquemático de una cámara CCD. El sistema óptico forma la imagen en el arreglo detector (CCD). La salida de este detector es una serie de pulsos analógicos que representa la intensidad de la imagen en una serie de elementos discretos.



*Figura 2.- Elementos de una cámara CCD*

Los tamaños típicos de arreglos de CCD varían desde 512x512 hasta 2048x2048 elementos. Cuando se usa una cámara analógica de este tipo, la señal de vídeo debe pasar a través de un convertidor análogo-digital (ADC) para que de esta manera sea posible almacenarla y procesarla en una computadora.

*Elementos para el almacenamiento de la imagen:* La unidad fundamental y más pequeña de memoria en un ordenador es el bit. La palabra "bit" es una contracción de *Binary Digit* (Digito Binario de su traducción del Inglés). La colección más importante de bits es el byte que consta de ocho bits y que tomándolos juntos representan otra unidad.

La cantidad de datos asociada con información visual (imágenes) es comúnmente grande. El almacenamiento de este tipo de información se mide en Kbytes (Mil bytes), Mbytes (millón de

Bytes), Gbytes (billón de Bytes) o Tbytes (Trillón de Bytes). Por ejemplo, una imagen de intensidad de 512 por 512 pixeles que requiere de un Byte de memoria por elemento, requiere de 0.27 Mbytes para ser almacenada.

El método más común para proveer almacenamiento de imágenes es la memoria de computadora (RAM). Otros dispositivos como los discos magnéticos y ópticos son también usados para el almacenamiento de imágenes digitales.

*Elementos para el procesamiento de la imagen* La parte medular en un sistema de procesamiento de imágenes es el procesamiento de la información visual que usualmente se expresa en forma de algoritmos computacionales, lo cual involucra la utilización e implementación de software especializado y el uso de computadoras personales o especializadas para llevará a cabo tareas específicas. Los fundamentos y tópicos esenciales para la implementación de algoritmos de procesamiento de imágenes son uno de los temas principales de la estancia técnica.

*Elementos de despliegue de imágenes:* Los monitores de TV o de computadora son los dispositivos mayormente usados para el despliegue de imágenes en sistemas de procesamiento. En los primeros años del procesamiento de imágenes el despliegue de imágenes también se llevó a cabo usando impresoras, principalmente para trabajos de procesamiento de baja resolución. El uso de caracteres de impresión se usó para simular los diferentes niveles de gris sobre la imagen.

## 2.- *VI's de adquisición.*

Los siguientes Vis son usados para crear y manipular imágenes, y para asignar y liberar las imágenes almacenadas en la memoria libre.



- *IMAQ Init* este VI, carga la configuración y la información de la cámara web. El **Interface Name** siendo el por default `img0` hace referencia al nombre para *MAX*. *IMAQ Int* genera una sesión en IMAQ y hace referencia a un dispositivo de adquisición específico para cualquier futura utilización de un NI-IMAQ driver.



- *IMAQ Close* detiene la adquisición, libera cualquier recurso almacenado, y cierra la específica sesión IMAQ de iniciada.



- *IMAQ Crea* una asignación en el buffer para una imagen para almacenar tu imagen.



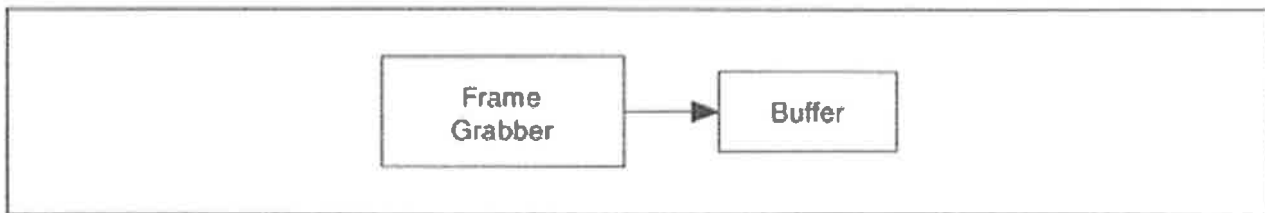
- *IMAQ Dispose* libera la memoria asignada en el buffer de memoria. Llama este VI solo después de que la imagen no va a ser requerida en el resto del proceso.

Una de las primeras cosas que se deben de hacer es determinar el modo de adquisición correcto para tú aplicación, se tiene que decidir si tu aplicación requiere una simple toma de una adquisición continua. Una adquisición simple solo corre una vez y luego se detiene, mientras que una adquisición continua corre indefinidamente hasta que el usuario detiene la adquisición.

Después, decide si tu aplicación requiere un *buffer* simple en memoria o varios *buffers*. Un buffer en memoria es un espacio utilizado para que tú mantengas tus imágenes.

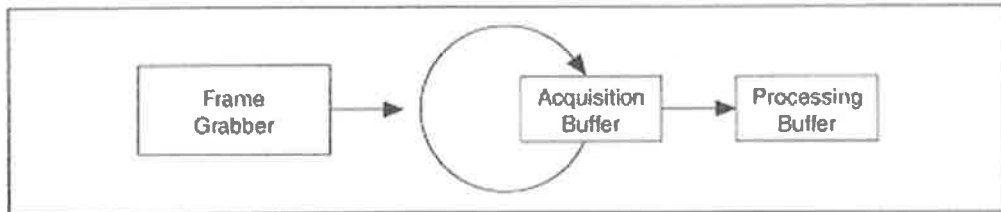
### 3.- *Buffer simple de adquisición mediante VI's*

El modo de adquisición mas simple es el instantaneo (*snap*), el cual solo adquiere una imagen simple en el buffer de memoria. La figura 3 ilustra la de una adquisición de instantánea (*snap*).



*Figura 3.- Adquisición Instantanea (Snap)*

Las adquisiciones tipo *Grab* transfieren imágenes a un buffer de memoria simple y sobrescribe ese mismo buffer con nuevas tomas a lo largo del proceso de adquisición. Los buffers que son copiados son necesarios para desplegar la imagen constantemente o solamente ser analizados puede ser de las dos formas. La siguiente figura ilustra una adquisición tipo grab.



*Figura 4.- Adquisición tipo Grab*

Usando la IMAQ Snap Vi, IMAQ Grab Setup, and IMAQ Grab Acquire Vi para adquirir las imágenes. Un Snap es una foto, si tú deseas adquirir una simple imagen de la cámara. Un grab es similar a un video, si tu lo deseas puedes adquirir cualquier imagen que venga de la cámara. Las imágenes en grab son mostradas secuencialmente, produciendo un video en tiempo real (Full-motion).



- IMAQ Snap adquiere una imagen simple en el buffer de memoria. Este Vi es para aplicaciones de baja velocidad o una captura simple.



- IMAQ Grab Setup Inicializa la adquisición y comienza la captura de la imagen en un buffer interno simple. Llama este Vi solo antes de utilizar un adquisición Grab.



- IMAQ Grab Acquire copia la imagen adquirida por el dispositivo hacia LabVIEW.

#### Mostrando las imágenes

Una de las primeras cosas que se desea hacer cuando se quiere adquirir una imagen es mostrarla en el monitor. NI Vision Vis posee una infinidad de caminos para mostrar las imágenes adquiridas.

## PROCEDIMIENTO

### *1. Inicialización de una Webcam e integración de sus librerías en LabVIEW*

Mediante el uso de una Web Cam, se hizo el procesamiento correspondiente de las imágenes, para ello es necesario instalar, controladores y plug-ins necesarios para que los dispositivos funcionen correctamente. Los pasos a seguir son:

- I) A continuación se despliega los archivos necesarios para que el software de NATIONAL INSTRUMENTS funcione correctamente y pueda detectar la cámara web, como un dispositivo para la adquisición de imágenes.
- II) Instalar **VideoOxe**, el cual es un active-x para poder utilizar video en LabVIEW, este lo encuentras en forma de prueba en la página web oficial del software.
- III) Instalar **NI\_USB\_IMAQ\_FOR\_USB**, la cual es una librería que es gratuita en la página web de soporte de NATIONAL INSTRUMENTS, los cuales son necesarios para que cree un submenú en LabVIEW donde viene cada VI, los VI que contiene esta librería son los siguientes: IMAQ USB snap.vi, IMAQ USB Grab Setup.vi, IMAQ USB Grab Acquire.vi, IMAQ USB Stop.vi, IMAQ USB Enumerate Cameras.vi, IMAQ USB Int.vi, IMAQ USB Close.vi, IMAQ USB Property Page.vi, IMAQ USB GetImageInfo.vi todos ellos necesarios para realizar la programación a bloques para el procesamiento de imágenes mediante la utilización de la Web Cam. Mas adelante se dará una explicación detallada de cual es la función de cada uno de estos vi's.
- IV) Una vez instalados es necesario Actualizar los controladores de DirectX, esto con la finalidad de que sean validas las actualizaciones que se hicieron al instalar el VIDEOOXE y el NI\_USB\_IMAQ\_FOR\_USB, sean detectadas completamente y detectados por LabVIEW.

A continuación se explica cada uno de los Vis NI\_USB\_IMAQ\_FOR\_USB, como se menciona anteriormente:

### *IMAQ USB Snap.vi*

Este Vi está diseñado para una simple adquisición de una toma simple, solo una cámara puede adquirir a un tiempo, mediante este VI utiliza la imagen de salida (Image Out) de para introducirla en una entrada de imagen.



IMAQ USB Snap.vi

### *IMAQ USB Grab Setup.vi*

Inicia una adquisición continua, una vez que la adquisición ha comenzado, llama al IMAQ Ubs Grab Acquire para copiar imágenes desde la continua adquisición, solo una cámara puede adquirir al momento.



IMAQ USB Grab Setup.vi

### *IMAQ USB Grab Acquire.vi*

Adquiere una imagen durante una adquisición continua, llama a IMAQ USB Grab Setup, para iniciar una adquisición continua. Solo una cámara puede adquirir a un tiempo, usando la salida de imagen de este vi se va hacia la entrada de imagen de IMAQ Create.vi



IMAQ USB Grab Acquire.vi

### *IMAQ USB Stop.vi*

Detiene una adquisición continua, IMAQ USB Grab Setup o IMAQ USB Snap, se encuentran antes de que se vaya a adquirir nuevas imágenes.



IMAQ USB Stop.vi

### *IMAQ USB Enumerate Cameras.vi*

Crea una lista de cámaras usb encontradas en el sistema. Utiliza la información de la lista con IMAQ USB Init para crear un IMAQ USB Session.



IMAQ USB Enumerate Cameras.vi

### *IMAQ USB Init.vi*

Crea una sesión IMAQ USB donde tomas el nombre de la cámara usb, que te proporciona IMAQ USB Enumerate Cameras las cuales están listas para iniciarlizarse. Las características del video para cámara Usb las determinas en el software de cada una de los equipos.



IMAQ USB Init.vi



### *IMAQ USB Close.vi*

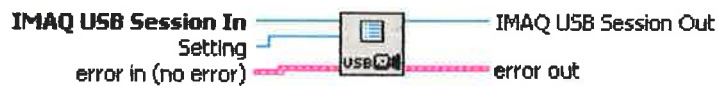
Cierra la sesión de la cámara usb que ha sido abierta con IMAQ USB Init.



IMAQ USB Close.vi

### *IMAQ USB PropertyPage.vi*

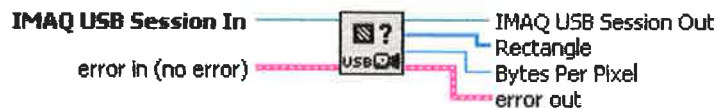
Este te proporciona la información del fabricante de la cámara y cualquier parámetro sobre la funcionalidad que tenga que ver directamente con el fabricante.



IMAQ USB PropertyPage.vi

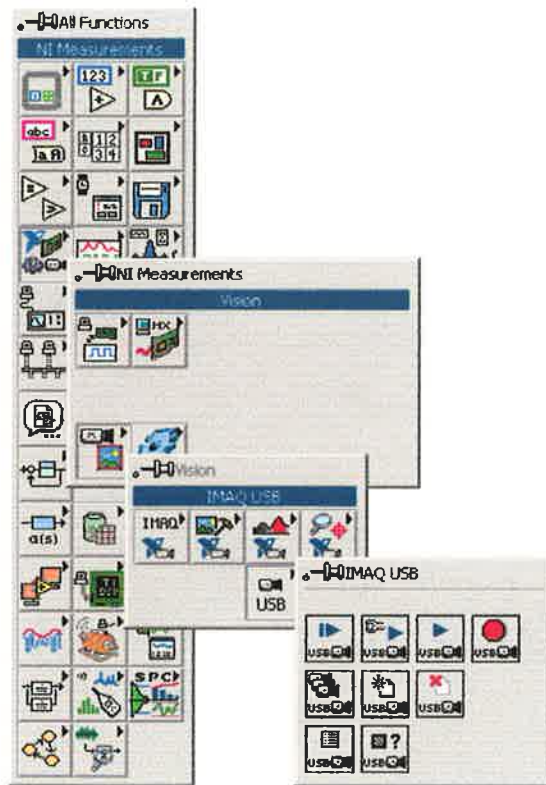
### *IMAQ USB GetImageInfo.vi*

Muestra toda la información acerca sobre la imagen de la sesión IMAQ USB

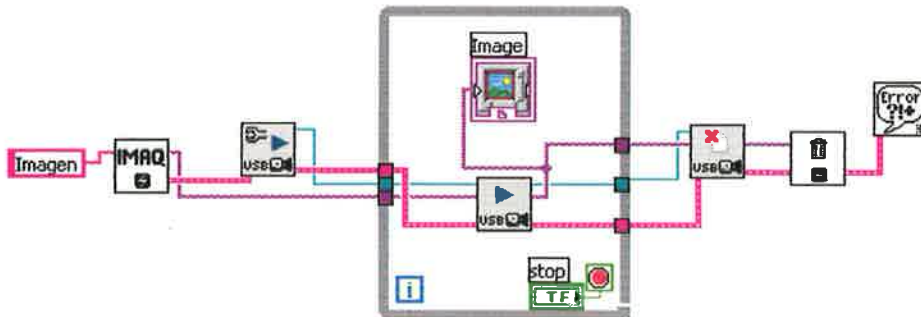


IMAQ USB GetImageInfo.vi

Estos Vis los puedes localizar dentro del menú de all functions en LabVIEW, a continuación se despliega una imagen que demuestra la ruta correcta de ubicación de ellos.



Una vez que ya tienes ubicados los vis con los cuales se obtendrá el procesamiento de imágenes se dispone a hacer algunas pruebas para configurar y ver si la web Cam esta funcionando correctamente un Vi de lo mas sencillo que solo es el que produce una imagen.



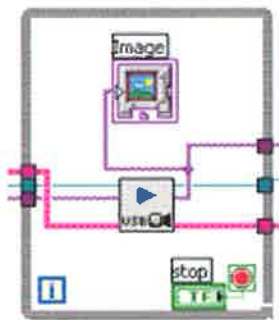
Vi de visualización de video en tiempo real

Como se puede apreciar en el vi anterior este se encarga de realizar una adquisición de imágenes en tiempo real, de inicio activas la sesión de la cámara esto con el fin de crear un espacio en memoria en la cual se estarán guardando temporalmente las imágenes, para luego ser desplegadas en pantalla, esto lo realizas con IMAQ.



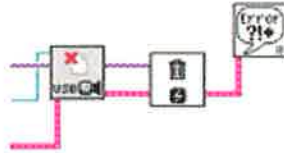
### IMAQ y IMAQ USB Grab Setup

Posteriormente se activa la cámara usb mediante el IMAQ USB Grab Setup.vi, estos dos vi se encuentran fuera del while loop, debido a que solo es necesario activarlos una vez, dentro del ciclo se encuentra lo que es en si el USB IMAQ Grab.vi, a su vez se encuentra el Display en el cual se despliega la imagen que esta siendo captada por la Web Cam en tiempo real, así como también se encuentra el botón de control Stop que es el que se encarga de detener las al ciclo cuando tu desees que la visualización de la imagen sea concluida.



### Adquisición y Procesamiento de la imagen

Una vez realizado esto se cierra la sesión de de la cámara web mediante el uso de IMAQ USB Close.vi, una vez liberado el dispositivo de adquisición de datos, se procede a liberar el buffer de memoria, donde las imágenes estaban siendo almacenadas antes de ser desplegadas.



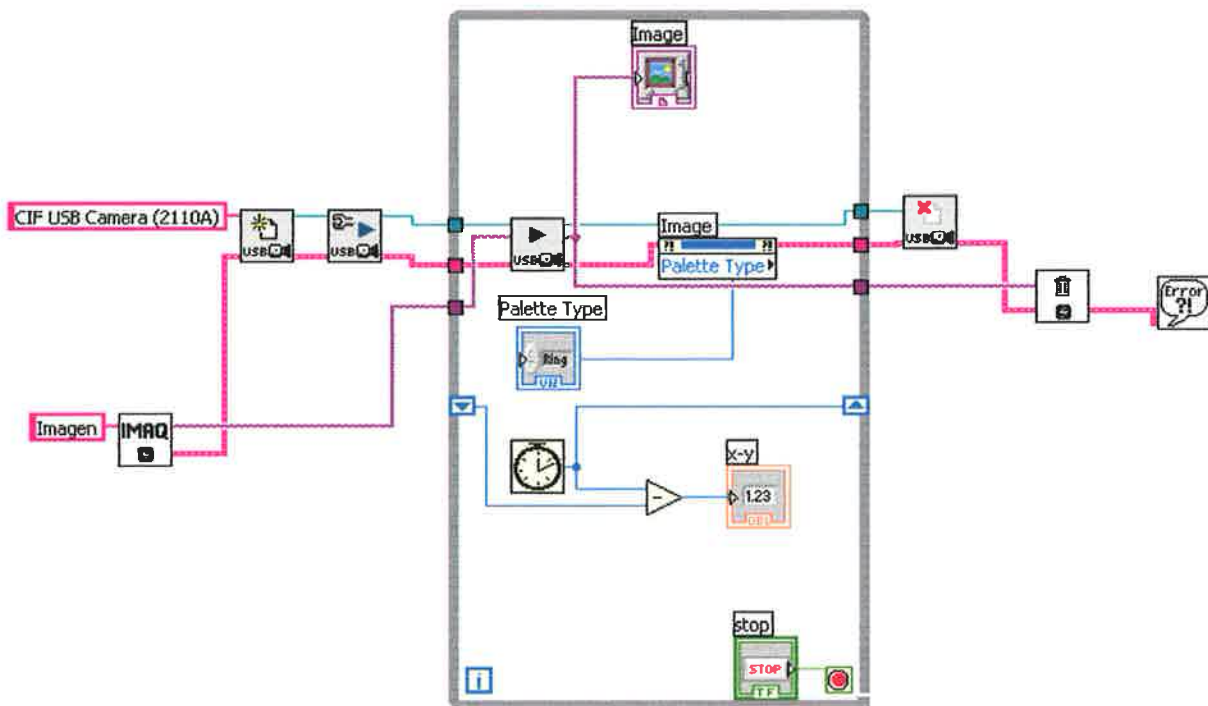
### Cerrar el dispositivo y liberando memoria

Debido a que siempre tienes que seguir un orden en la adquisición de imágenes fue que se optó por realizar sub-Vis los cuales te evitan hacer cierto tipo de trabajo repetitivo, acelerando el proceso de programación para cuando desees hacer en su momento.

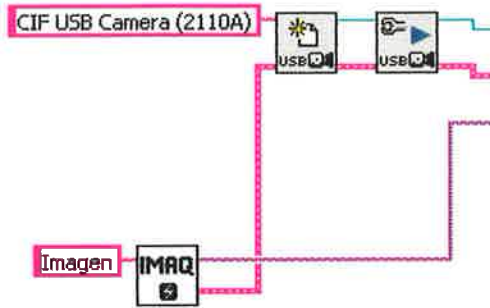
Una vez entendido los conceptos de cómo hacer una captura y de cómo visualizar un video en tiempo real por medio de un dispositivo de video Web Cam se procede a realizar un poco mas detallado el procesamiento de imágenes, como obtener información de la imagen, realizar una función inversa de una imagen, conversión a 8 bits, etc.

### Creación de Sub-Vi Palette Type.vi

La creación de este sub-Vi nos llevara a un diseño conceptual de cómo podemos obtener información de la imagen mediante el comando *palette type*, el cual te permite saber la información de la imagen que estas visualizando en su momento, lo primero que se hizo fue realizar el Vi general el cual se encarga del proceso en general, una vez realizado el vi general se procede a la parte que es el procesamiento convertirlo en un Sub-Vi, de forma que el Vi general se concluyo como lo demuestra el Vi siguiente:



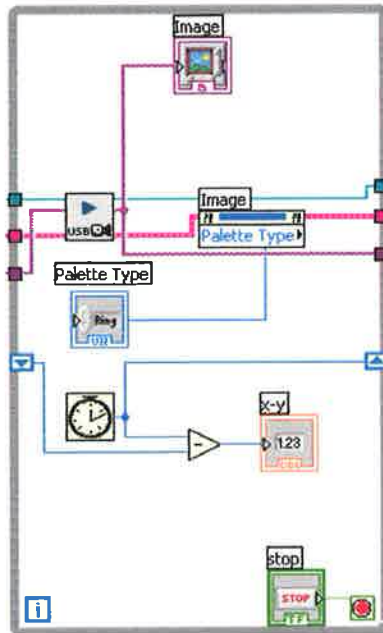
De cómo en todo programa de adquisición de datos se tiene que seguir la regla general para el procesamiento de imágenes, activando la primer parte IMAQ, después inicializar el dispositivo de adquisición de imágenes, mediante IMAQ USB Init, al cual se le tiene que asignar el nombre de la cámara que estas utilizando esto con el fin de decirle que utilicé dicho driver para realizar la adquisición de la imagen, así mismo la operación a realizarse es visualizar el video en tiempo real, por lo tanto utilizas el IMAQ USB Grab Setup, a continuación se visualiza cual la parte inicial.



Asignación en memoria, activación de la cámara y inicialización de la cámara

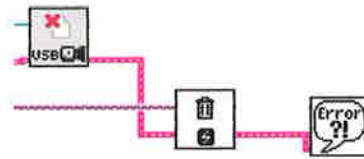
Dentro del procesamiento por ser en tiempo real se utiliza el ciclo while loop, que es el que mejor se implementa en este tipo de situaciones ya que te da la opción de tu poder interrumpirlo en el momento que lo desees, dentro del ciclo se encuentra lo que es procesamiento a realizar, en este caso utilizar el comando Palette Type para que este te de la información de la imagen como es el tamaño, las el corrimiento, entre otras.

La siguiente figura demuestra esta etapa.



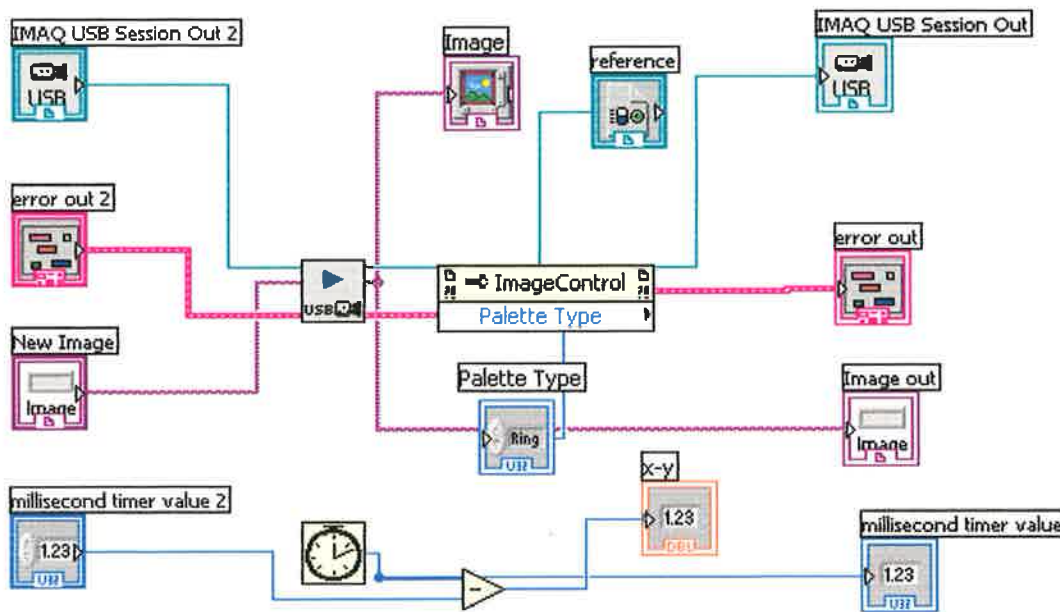
Etapa de procesamiento

Y por ultimo hay que finalizar la sesión y así para liberar el dispositivo y la memoria interna del buffer.



### Etapa de Finalización

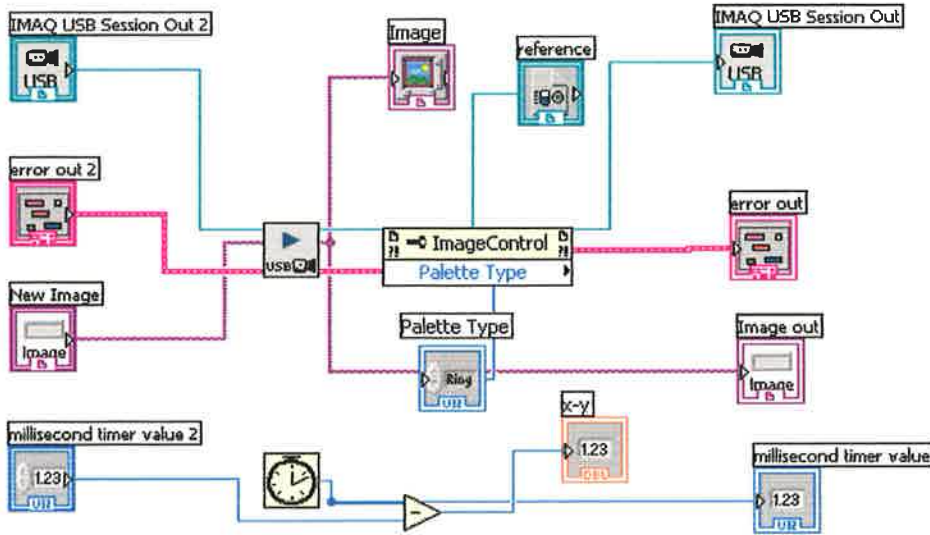
Con el fin de reducir el tiempo de procesamiento y el tiempo de implementación en el diseño se elaboro la librería correspondiente a este procesamiento, quedando de la siguiente forma:



### Sub Vi correspondiente al proceso de Información

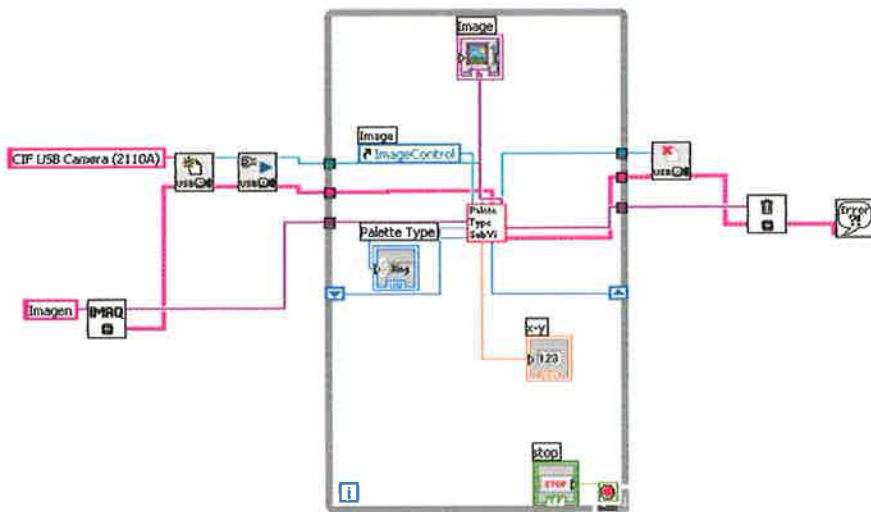
Como se puede visualizar en la imagen esta tiene un grupo de entrada y otro de salida siendo de izquierda a derecha correspondientemente, la primera en el cual entra la señal que indica que el dispositivo de adquisición de imágenes se encuentra encendido y listo para comenzar a adquirir las imágenes, el error, la señal de imagen en la cual es el espacio en memoria que se

ha asignado a ella, así como un conector de tiempo que es implementado en el ciclo, siendo estas las entradas a su vez se tiene salidas también, las cuales son exactamente las mismas para poder llevar a acabo el ultimo paso de la adquisición de imágenes, la finalización. Quedando el sub-vi de la siguiente forma:



Sub-Vi *Palette Type*

Una vez realizado el Sub-Vi se procede a implementarlo dentro del vi General, siendo mostrado en el siguiente diagrama:



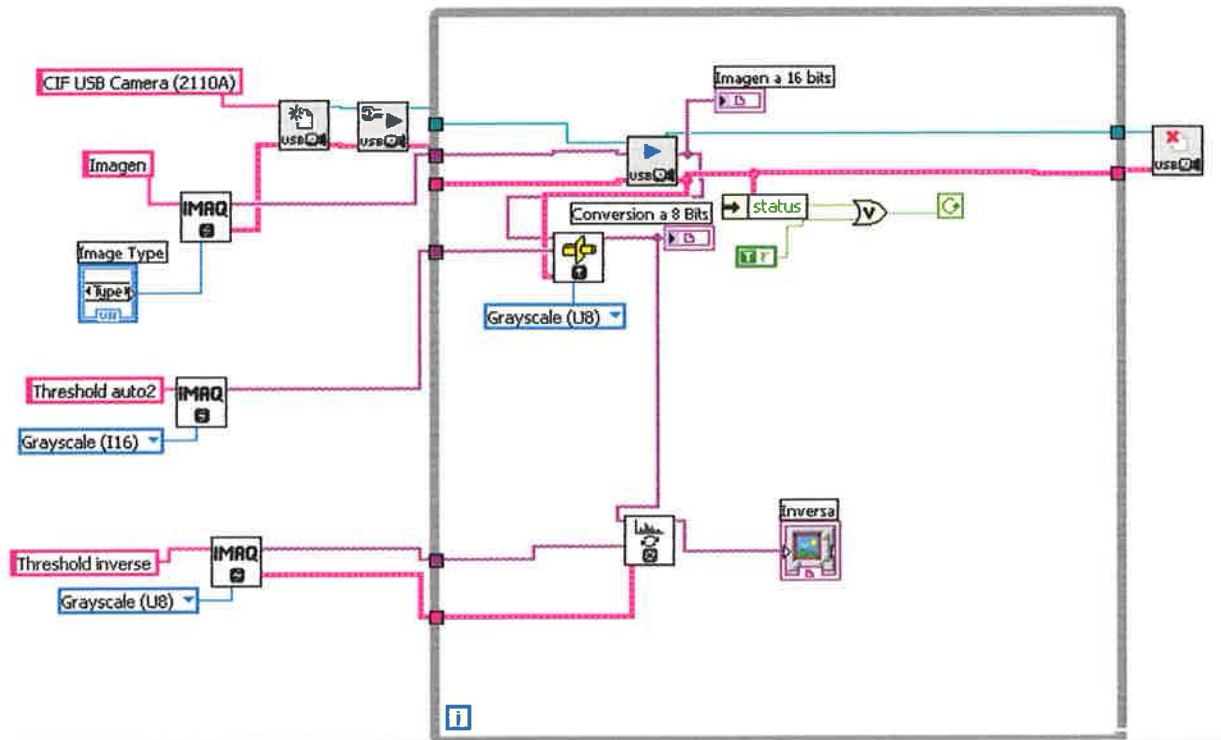
Procesamiento de imagen utilizando el Sub-Vi *Palette Type*



### Creación de sub-vi, conversor a 8 bits y negativo

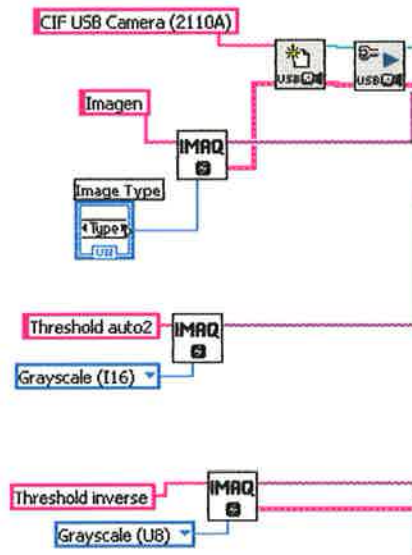
Otro proceso en el cual se trabajo fue ya con una imagen en si modificando sus diferentes niveles de tono, así como la conversión a 8 bits y la función inversa de los colores, estas tres actividades se desplegaron en un solo sub-vi del cual se podrá utilizar cualquiera que sea la función que quieras necesites usar de ellas, solo con utilizar o no la terminal de salida que se necesite en el momento.

De igual forma la creación de sub-vi anterior fue la misma para este, siguiendo los pasos necesarios para la adquisición de imágenes, diseñándolo de forma general para el procesamiento a realizar y por ultimo creando el sub-vi correspondiente el cual recibe el nombre de Inversa\_Image.vi, el cual quedo realizado en forma general de la siguiente manera:



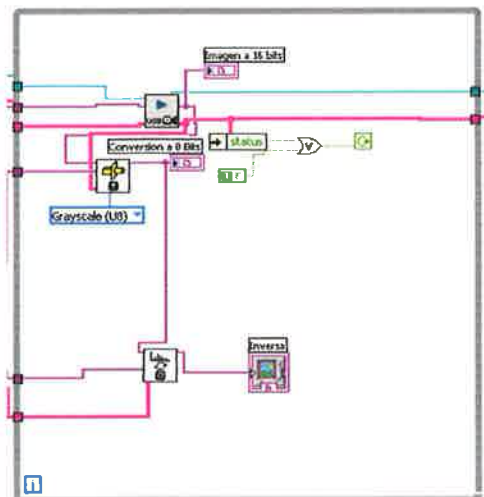
Vi general del procesamiento de imagen, conversor a 8 bits y función inversa

En este caso al inicializar fue necesario crear 3 espacios en el buffer memoria debido a que son ahora 3 imágenes las que se están desplegando, las cuales son la imagen original, la imagen convertida a 8 bits, y la imagen inversa en colores, la etapa inicial de este Vi se muestra a continuación.



Inicialización del equipo y asignaciones en el buffer de memoria

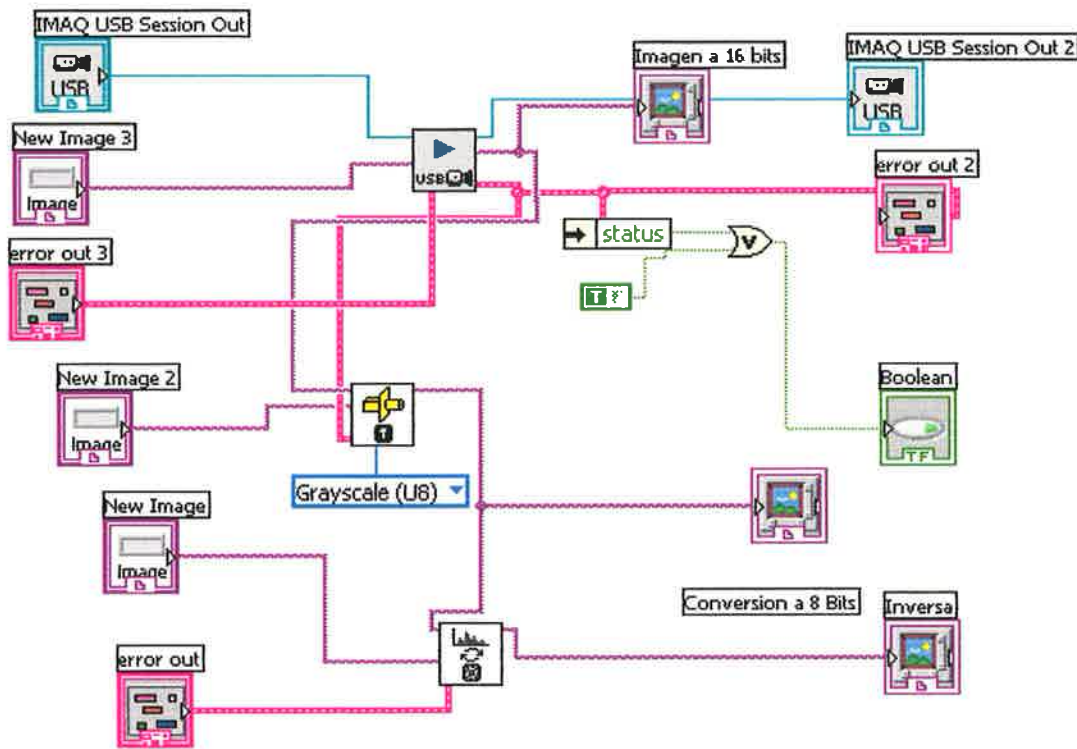
De igual forma el procesamiento de las imágenes se da en la parte intermedia del sistema, por lo tanto el procesamiento en este caso corresponde a la siguiente figura:



Procesamiento, conversión a 8 bits e inversión de colores

De igual forma para cumplir con las especificaciones dadas en el formato de adquisición de imágenes se concluye cerrando el dispositivo y liberando memoria.

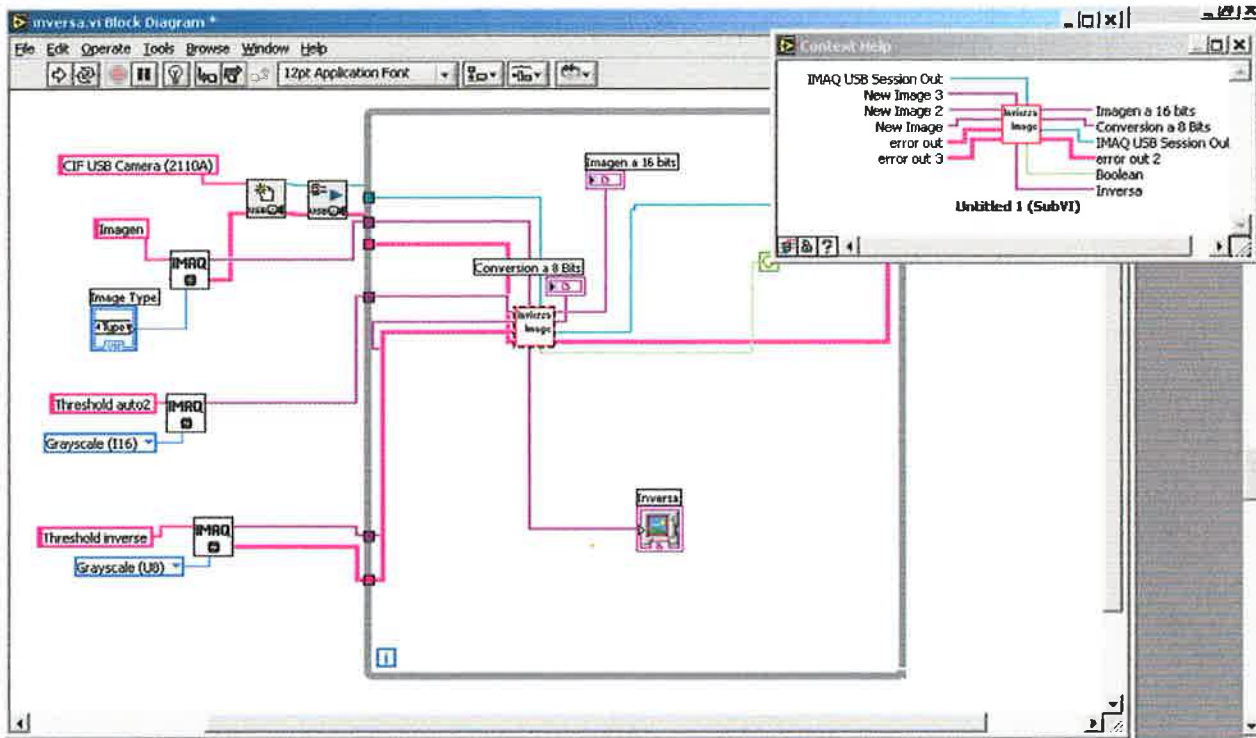
Una vez terminado en general se conduce a realizar el sub-vi correspondiente, el cual quedo conformado de la siguiente manera:



Sub-Vi conversor a 8 bits e inversor

Este Vi posee 6 entradas de las cuales es la obtención de la imagen correspondiente a la inicialización del dispositivo, como también las 3 asignaciones de memoria correspondientes para desplegar las imágenes en sus correspondientes displays, en su parte intermedia posee lo que es IMAQ USB Grab.vi el cual se explico en los vis pasado su función, también se encuentra el conversor a 8 bits, y el inversor de imágenes, como salidas están los displays, un botón condicional, como también la salida del video y el error.

La siguiente imagen muestra el Vi, ya con su subvi utilizado:



Procesamiento de imágenes usando un Sub-vi

A continuación se muestran las 3 imágenes después del procesamiento dado por el vi.



Imágenes Procesadas

## **CONCLUSIONES**

Se cumplieron satisfactoriamente los objetivos planteados al inicio del proyecto, donde las librerías creadas podrán ser utilizadas para diferentes tipos de aplicaciones, también simplificará el diseño de las próximas apelaciones a realizar.

Así como también se conoció más a fondo las aplicaciones que se pueden llegar a realizar en estas dos aplicaciones de National Instruments, software como LabVIEW y Vision Assistant.

## **BIBLIOGRAFIA**

- IMAQ Vision for LabView, User Manual, National Instruments, June 2003, Part Number 32291B-01.
- Manual de Usuario, Curso de LabView, National Instruments, México, 2001.
- IMAQ Vision for concepts, Manual, National Instruments, June 2003.  
LabView User Manual, National Instruments, April 2003, Part Number 320999E-01.