

UNIVERSIDAD TÉCNICA DE COTOPAXI



CARRERA DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

INGENIERÍA EN INFORMÁTICA Y SISTEMAS
COMPUTACIONALES

TESIS DE GRADO

TESIS PRESENTADA PARA LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN INFORMÁTICA Y SISTEMAS
COMPUTACIONALES

TEMA:

“IMPLEMENTACIÓN DE UN SOFTWARE PROTOTIPO DE PROCESAMIENTO Y EXTRACCIÓN DE RASGOS, PARA MEJORAR LA CALIDAD DE LA IMAGEN; UTILIZANDO TÉCNICAS DE INTELIGENCIA ARTIFICIAL, PARA LA ESPECIALIZACIÓN DE DISEÑO GRÁFICO DE C.I.Y.A., EN EL PERIODO NOVIEMBRE 2007-JULIO 2008”.

Autores:

Chicaiza Toapanta Daniel Darío.

Falconí Hidalgo Renato Danilo.

Director:

Ing. Franklin Javier Montaluisa.

LATACUNGA – ECUADOR

Agosto 2008

INFORME FINAL DEL DIRECTOR DE TESIS

Cumpliendo con lo estipulado en el capítulo IV Art.26, literal h) del Reglamento de Graduación de Nivel de Pregrado de la Universidad Técnica de Cotopaxi, informo que el grupo conformado por: Chicaiza Toapanta Daniel Darío, y Falconí Hidalgo Renato Danilo, egresados de Ingeniería en Informática y Sistemas Computacionales; han desarrollado su investigación de grado de acuerdo a los planteamientos formulados en el proyecto de Tesis.

En virtud de lo antes expuesto considero que el grupo se encuentra habilitado para presentar el acto de defensa de Tesis acerca de la: “IMPLEMENTACIÓN DE UN SOFTWARE PROTOTIPO DE PROCESAMIENTO Y EXTRACCIÓN DE RASGOS, PARA MEJORAR LA CALIDAD DE LA IMÁGEN; UTILIZANDO TÉCNICAS DE INTELIGENCIA ARTIFICIAL, PARA LA ESPECIALIZACIÓN DE DISEÑO GRÁFICO DE C.I.Y.A., EN EL PERIODO NOVIEMBRE 2007- JULIO 2008”.

Latacunga, agosto del 2008.

“POR LA VINCULACION DE LA UNIVERSIDAD CON EL PUEBLO”

DIRECTOR

Ing. Franklin Javier Montaluisa.

C.I: 050216679-6

AUTORÍA DE TESIS

Los abajo firmantes, en calidad de egresados de la Universidad Técnica de Cotopaxi, Carrera de Ciencias de la Ingeniería y Aplicadas, Especialización Ingeniería en Informática y Sistemas Computacionales, declaramos que los contenidos de esta Tesis de grado, requisito previo a la obtención del Título de Ingeniero en Informática y Sistemas Computacionales, son absolutamente originales, auténticos, personales y de exclusiva responsabilidad legal y académica de los autores.

Latacunga, agosto del 2008.

Chicaiza Toapanta Daniel Darío
C.I. 050240710-9

Falconí Hidalgo Renato Danilo
C.I. 050297078-3

APROBACIÓN DEL TRIBUNAL DE GRADO

La Tesis de grado, ha sido revisada, aprobada y autorizada su impresión y empastado; previo a la obtención del Título de Ingeniero en Informática y Sistemas Computacionales; por lo tanto autorizamos a los autores a la presentación a efectos de su sustentación pública.

Latacunga, agosto del 2008.

El Tribunal

Dra. Msc Anita Chancusi
PRESIDENTA DEL TRIBUNAL

Ing. Fabián Masapanta
OPOSITOR DEL TRIBUNAL

Ing. Mario Banda
MIEMBRO DEL TRIBUNAL

DEDICATORIA

Dedicamos este material investigativo, a Dios por ser nuestro creador y el que actúa en la trayectoria de nuestras vidas. A nuestros padres, en especial a nuestras madres; por ser una amiga, confidente y la que nos han apoyado en todas las circunstancias. A nuestras familias por aportar en nuestra vida con una idea, un consejo, su compañía y su cariño. A los docentes de nuestra querida Universidad Técnica de Cotopaxi, en particular ha aquellos de Ingeniería en Informática y Sistemas Computacionales que día a día, semestre a semestre inculcaron en nosotros conocimientos científicos y morales. A nuestro director Ing. Franklin Javier Montaluisa que con su amistad, con su guía y entrega educativa; permitieron un caudal apropiado en el desarrollo de este proyecto.

Esperamos que al cumplir esta meta, recompensar en alguna forma todo el sacrificio y apoyo que nos han brindado.

Daniel Darío Chicaiza Toapanta – Renato Danilo Falconí Hidalgo

AGRADECIMIENTO

En la historia está comprobado que no ha existido ni existirá ningún ser humano que pueda sobrevivir individualmente; mucho menos sobresalir y desarrollarse íntegramente. Por lo que agradecemos el éxito de este proyecto a todos aquellos que aportaron de una u otra manera con la finalización del mismo. A la Universidad Técnica de Cotopaxi, de sobremanera al apoyo y la ayuda sincera de nuestro director de tesis: Ing. Franklin Javier Montaluisa.

Estamos seguros que todo este cúmulo de conocimientos adquiridos, sabremos poner al servicio de quienes lo necesiten. Con una vida profesional a favor de la Patria y la sociedad.

Daniel Darío Chicaiza Toapanta – Renato Danilo Falconí Hidalgo

ÍNDICE GENERAL

CONTENIDOS	PÁGINAS
PORTADA.....	i
INFORME FINAL DEL DIRECTOR DE TESIS	ii
AUTORÍA DE TESIS	iii
APROBACIÓN DEL TRIBUNAL DE GRADO	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE GENERAL.....	vii
ÍNDICE DE CUADROS.....	xi
ÍNDICE DE GRÁFICOS	xi
ÍNDICE DE FIGURAS.....	xii
RESUMEN	xv
ABSTRACT	xvi
INTRODUCCIÓN	1
CAPÍTULO I	6
“EL SER HUMANO Y LA VISIÓN POR COMPUTADORA”	6
1.1 Introducción	6
1.2 El Sistema de Visión Humano	7
1.2.1 El Ojo	8
1.3 Sistema de Visión de Computadora	10
1.3.1 Cámaras.....	10
1.3.2 Interfaces de Computadora	12
1.3.3 Procesando una Imagen	13
1.4 Herramientas Matemáticas.....	14
1.5 MATLAB.....	15
1.5.1 Origen de MATLAB	15
1.5.2 ¿Qué es MATLAB?	16
1.5.3 Plataformas, Requerimientos y Estructura de MATLAB	17
1.5.4 Utilidades y Funcionalidades	19
1.6 Algunos Comandos y Herramientas de MATLAB.....	20

1.6.1 Manipulación de Vectores y Matrices.....	21
1.7 Imágenes, Probando y Procesando.....	21
1.7.1 Introducción	21
1.7.2 Transformación de Fourier.....	34
1.7.3 Histogramas	36
1.8 Extracción de Rasgos	36
1.9 Introducción a la Textura y Segmentación.....	37
1.9.1 Textura	37
1.9.2 Segmentación	38
CAPITULO II	40
“DESCRIPCIÓN, ANÁLISIS E INTERPRETACIÓN DE RESULTADOS”	40
2.1 Breve descripción de la Universidad Técnica de Cotopaxi	40
2.1.1 Descripción de la Especialización de Diseño Gráfico	41
2.1.2 Análisis de la Importancia de la Imágenes en Diseño Gráfico	42
2.1.2.1 Formatos de gráficos más importantes.....	42
2.1.2.2 Fundamentos del color	44
2.1.2.3 Modelos de color.....	44
2.1.2.4 Mapa de color.....	45
2.2 Aplicaciones sobre el Procesamiento de Imágenes.....	46
2.2.1 Rayos gama	47
2.2.2 Rayos X.....	48
2.2.3 Ultravioleta.....	49
2.2.4 Visible e Infrarroja	50
2.2.5 Microondas.....	52
2.2.6 Ondas de Radio	53
2.2.7 Aplicaciones de otras Fuentes de Energía.....	53
2.3 Análisis de resultados de la encuesta desarrollada por los Estudiantes y Docentes de Diseño Gráfico.....	54
2.3.1 Verificación de la Hipótesis	64
2.4 Posibles funcionalidades, y ventajas para la solución de problemas con respecto a las imágenes en la especialización de Diseño Gráfico.....	65

2.5 Esquemas técnicos y figuras en el desarrollo del software prototipo para el Procesamiento de Imágenes y Extracción de Rasgos	68
2.5.1 Introducción	68
2.5.2 Pasos Fundamentales del Procesamiento Digital de Imágenes.....	70
2.5.3 Componentes de un sistema de PDI de propósito general	72
CAPITULO III	75
“FACTIBILIDAD EN LA IMPLEMENTACIÓN DE UN SOFTWARE PROTOTIPO DE PROCESAMIENTO Y EXTRACCIÓN DE RASGOS, PARA MEJORAR LA CALIDAD DE LA IMAGEN; UTILIZANDO TÉCNICAS DE INTELIGENCIA ARTIFICIAL, PARA LA ESPECIALIZACIÓN DE DISEÑO GRÁFICO DE C.I.Y.A.”	75
3.1 Presentación	75
3.2 Justificación.....	77
3.3 Objetivos	78
3.4 Factibilidad de aplicar el software prototipo, en la especialización de Diseño Gráfico.....	80
3.4.1 Introducción	80
3.4.2 Ruido en imágenes	80
3.4.3 Escala de grises	81
3.4.4 Imagen binaria.....	81
3.4.5 Imagen Indexada	82
3.4.6 Imagen RGB	82
3.4.7 Imagen multi-cuadro	82
3.5 Impacto y vida útil de software prototipo	82
3.6 Desarrollo del software prototipo	83
3.6.1 Procesamiento de Imágenes	85
3.6.1.1 Procesamiento básico de imágenes	86
3.6.1.2 Extracción de rasgos	89
3.6.1.3 Información final sobre Visión Artificial.....	91
3.7 Elaboración del Manual para el Usuario.....	93
3.8 GUI con MATLAB	112
3.8.1 Cómo crear aplicaciones GUI con GUIDE.....	112

3.8.2 Archivos generados por GUIDE	114
3.8.3 Estableciendo las propiedades de los componentes del GUI.....	115
3.8.4 Administrando datos del GUI con Estructuras de Handles.....	119
“CONCLUSIONES Y RECOMENDACIONES”	122
CONCLUSIONES	122
RECOMENDACIONES	124
REFERENCIAS BIBLIOGRÁFICAS.....	125
Bibliografía Citada	125
Bibliografía Consultada	125
Bibliografía Virtual	126
ANEXOS	127
ANEXO N° 1.....	128
ANEXO N° 2.....	131

ÍNDICE DE CUADROS

Cuadro N° 1.0: Formatos y extensiones soportadas por MATLAB.....	23
Cuadro N° 2.0: Definición de imagen digital	55
Cuadro N° 3.0: Importancia del PDI en Diseño Gráfico.....	56
Cuadro N° 4.0: Entendimiento acerca del procesamiento de imágenes.....	57
Cuadro N° 5.0: Definición de Computadora	58
Cuadro N° 6.0: Procesamiento de Imágenes Digitales.....	59
Cuadro N° 7.0: Destreza para maniobrar programas.....	60
Cuadro N° 8.0: Uso de un software para procesar imágenes	61
Cuadro N° 9.0: Software para procesar imágenes digitales	62
Cuadro N° 10.0: Manual de Uso	63

ÍNDICE DE GRÁFICOS

Gráfico N°. 1: Definición de imagen digital	55
Gráfico N°. 2: Importancia del PDI en Diseño Gráfico	56
Gráfico N°. 3: Entendimiento acerca del procesamiento de imágenes	57
Gráfico N°. 4: Definición de Computadora.....	58
Gráfico N°. 5: Procesamiento de Imágenes Digitales	59
Gráfico N°. 6: Destreza para maniobrar programas	60
Gráfico N°. 7: Uso de un software para procesar imágenes.....	61
Gráfico N°. 8: Software para procesar imágenes digitales.....	62
Gráfico N°. 9: Manual de Uso.....	63

ÍNDICE DE FIGURAS

Figura 1.0: El Ojo Humano	8
Figura 2.0: Diferencia entre una cámara convencional y una cámara digital	11
Figura 2.1: Clasificación de las Cámaras	11
Figura 3.0: Interfaz de computadora	13
Figura 4.0: Estructura de MATLAB.	18
Figura 5.0: Representación de una imagen a escala de grises en MATLAB	23
Figura 5.1: Representación de una imagen a color RGB en MATLAB	24
Figura 5.2: Obtención del valor de un píxel.....	25
Figura 5.3: Planos de la imagen a) rojo, b) verde y c) azul.....	25
Figura 5.4: Ejemplo de sub-muestreo	26
Figura 5.5: Filtrado espacial por una máscara de 3 x 3.....	28
Figura 5.6: Imagen resultado del filtraje espacial.	29
Figura 5.7: Imagen resultado de la aplicación del algoritmo de canny.....	29
Figura 5.8: Imagen resultado de la aplicación de un umbral	30
Figura 5.9: Imagen resultado de la aplicación de la operación morfológica	31
Figura 5.10: Imagen binaria conteniendo un objeto.....	32
Figura 5.11: Ejemplo de imagen indexada.....	32
Figura 5.12: Identificación de los centroides	33
Figura 5.13: Imagen RGB	33
Figura 5.14: Imagen HSV	33
Figura 6.0: Suma de los colores primarios.....	44
Figura 6.1: Espacio del modelo de color RGB	45
Figura 7.0: Espectro EM en energía por fotón.....	47
Figura 8.0: Tomografía por emisión de positrones (PET)	47
Figura 8.1: Imagen a partir de rayos gamma de Cygnus Loop.....	48
Figura 9.0: Angiograma de la aorta.....	48
Figura 9.1: Tomografía de la cabeza	49
Figura 10.0: Ejemplos de imágenes ultravioleta	49
Figura 11.0: Superficie de un CD de audio al microscopio	50
Figura 11.1: Bandas del satélite LANDSAT.....	51
Figura 11.2: Control de calidad.....	52

Figura 12.0: Imagen de radar del tibet.	52
Figura 13.0: Imágenes MRI de una rodilla y una espina.	53
Figura 14.0: Ejemplo de rectificación de perspectiva.	65
Figura 14.1: Ejemplo de rectificación de distorsión de lente.	66
Figura 14.2: Triangulación: estimación de Q a partir de A y B.	66
Figura 14.3: Correspondencia en tres puntos.	67
Figura 15.0: Esquema general de Inteligencia Artificial, y de la visión.	68
Figura 15.1: Proceso de desarrollo iterativo controlado.	69
Figura 16.0: Elementos de un sistema de visión por computadora.	70
Figura 16.1: Pasos del PDI.	72
Figura 16.2: Componentes de un sistema de PDI de propósito general.	73
Figura 17.0: Visión por Computadora del software SOWMEIM v1.0.	84
Figura 17.1: Funcionalidad del software SOWMEIM v1.0.	84
Figura 17.2: Interfaz del software prototipo SOWMEIM v1.0.	85
Figura 18.0: Procesamiento de bajo nivel.	85
Figura 18.1: Procesamiento de nivel medio.	86
Figura 18.2: Procesamiento de alto nivel.	86
Figura 19.0: Diagrama en bloques del procesamiento de imágenes.	87
Figura 19.1: Imagen de Jesús, Imagen RGB, Escala grises, Blanco y Negro.	87
Figura 19.2: Imagen de una caricatura, Imagen aplicada un filtro Pasa bajas.	87
Figura 19.3: Imagen aplicada un filtro Pasa altas, y un filtro Pasa bandas.	88
Figura 19.4: Imagen de la biblioteca de MATLAB, Imagen binarizada.	88
Figura 19.5: Imagen de la biblioteca de MATLAB, aumentada el brillo.	88
Figura 20.0: Ejemplo de una región, Imagen, Región segmentada.	89
Figura 20.1: Imagen original, Imagen en bordes.	89
Figura 20.2: Imagen original, Segmento de la imagen.	90
Figura 20.3: Imagen original, Bordos, Segmento, Rasgos de la imagen.	90
Figura 21.0: Dispositivos para captura de imágenes.	92
Figura 21.1: Proceso principal en Visión por computadora.	92
Figura 22.0: Carátula del manual del usuario.	93
Figura 23.0: Interfaz gráfica (SOWMEIM v1.0).	100
Figura 24.0: Rutinas del menú Imagen.	100

Figura 24.1: Ventana del ítem Abrir.	101
Figura 24.2: Ítem Salir	102
Figura 25.0: Rutinas del menú Edición.....	102
Figura 26.0: Rutinas del menú Conversiones.	104
Figura 26.1: Botones de conversiones de imagen.....	104
Figura 27.0: Rutinas del menú Máscara.....	105
Figura 28.0: Rutinas del menú Tratamiento.....	106
Figura 28.1: Rutinas del menú Filtros.....	107
Figura 28.2: Rutinas del menú Extracción de bordes.....	108
Figura 29.0: Rutinas del menú Histograma.	108
Figura 30.0: Ventana del menú Ayuda	109
Figura 31.0: Deslizadores de control	110
Figura 32.0: Ventana Extracción de Rasgos	110
Figura 33.0: Ventana Capturar Imagen.....	111
Figura 34.0: Ventana Aplicaciones del Procesamiento de Imágenes	111
Figura 35.0: GUIDE.....	113
Figura 35.1: Editor de propiedades	116
Figura 35.2: Editando Callbacks	117
Figura 35.3: Asignación del tag de un objeto	117

“IMPLEMENTACIÓN DE UN SOFTWARE PROTOTIPO DE PROCESAMIENTO Y EXTRACCIÓN DE RASGOS, PARA MEJORAR LA CALIDAD DE LA IMÁGEN; UTILIZANDO TÉCNICAS DE INTELIGENCIA ARTIFICIAL, PARA LA ESPECIALIZACIÓN DE DISEÑO GRÁFICO DE C.I.Y.A., EN EL PERIODO NOVIEMBRE 2007-JULIO 2008”.

RESUMEN

Esta tesis describe la herramienta SOWMEIM v1.0, que hace parte del gran tema Visión Computacional. El diseño efectuado es un software prototipo que utiliza las Cajas de Herramientas o "Toolboxes" de MATLAB, para Procesamiento y Adquisición de imágenes; para conectar una cámara a la computadora creando una imitación básica del sistema de visión del ser humano, capaz de manejar información relacionada con imágenes del mundo real. La información provista, así como las imágenes, pueden estar localizadas en algún sitio del disco duro. Estos recursos incluyen componentes tanto de hardware (cámara) como de software (ej. MATLAB). El software prototipo SOWMEIM v1.0 permite el procesamiento de imágenes de forma distribuida y realiza el mejoramiento de imágenes de acuerdo a las especificaciones del usuario. Procesada la imagen, SOWMEIM v1.0 esta en capacidad de: guardar la imagen resultado, recolectar imágenes en diferente formato y ofrecer al cliente nuevas formas de interpretación de sus imágenes. La arquitectura del software prototipo SOWMEIM v1.0 y los algoritmos usados para el diseño de la herramienta se discutirán en la fundamentación teórica y metodológica de la tesis.

ABSTRACT

This thesis describes the tool SOWMEIM v1.0 that makes part of the great topic Computer Vision. The effected design is a prototype software that uses the MATLAB's Toolboxes, for processing and acquisition of images; for connecting a camera to the computer creating a basic imitation of the vision system of the human being, able to manage information related with images of the real world. The provided information, as well as the images, can be located in some place of the hard disk. These resources include both components, hardware (camera) and software (e.g. MATLAB). The prototype software SOWMEIM v1.0 allows the processing of images in a distributed way and it carries out the improvement of images according to the user's specifications. Processed the image, SOWMEIM v1.0 is in capacity of: store the result image, to gather images in different format and to offer to the client new forms of interpretation of their images. The architecture of the prototype software SOWMEIM v1.0 and the algorithms used for the design of the tool will be discussed in the theoretical and methodological foundation of the thesis.

INTRODUCCIÓN

Uno de los mayores obstáculos en el procesamiento de imágenes obtenidas con la utilización de cámaras y sensores digitales, es la carencia de herramientas eficientes y fáciles de utilizar para procesar y, a su vez mejorar la apariencia visual de la imagen. Procesar información de manera local requiere de un software rápido y en ocasiones a pesar de disponer de estas herramientas no siempre es factible realizar el procedimiento, pues manipular exagerada cantidad de información implica aumentar el tiempo de ejecución y por tanto elevar el costo computacional y “perder la paciencia en alguna que otra tarea”. Afortunadamente, la tecnología de hoy ofrece la posibilidad de crear software y aplicaciones capacitados para hacer frente a estos inconvenientes, aprovechando los recursos de cómputo que pueden existir en universidades y centros de investigación. Por otra parte, la existencia imágenes de baja calidad, distorsionadas, o degradadas dispersas alrededor del mundo ha sido objeto de preocupación por parte de los investigadores que trabajan en esta área.

Para tratar algunos de estos inconvenientes, la especialidad en Ingeniería en Informática y Sistemas Computacionales de la Universidad Técnica de Cotopaxi a través de sus tesis de investigación nos ha permitido incursionar en el desarrollo de un software para procesar imágenes, extraer los rasgos y mejorar su calidad; que es el principio básico de Visión por Computadora, ya que la imagen es captada por medio de una cámara. La infraestructura que utilizara este software prototipo es la especialización en Diseño Gráfico, misma que será capaz de procesar imágenes de manera sencilla.

El software prototipo SOWMEIM v1.0 correlaciona en forma eficiente herramientas software y hardware, mostrando a los usuarios un software homogéneo. Además proporciona un método de interacción usuario – imagen, más directo o más real.

Esta herramienta fue desarrollada para mejorar la calidad de las imágenes, de ahí su nombre, que son las iniciales de su funcionalidad SOW (Software) ME (Mejoramiento) IM (Imágenes) (v1.0) (Versión 1.0).

Como pronunciamos en las líneas anteriores las funciones del SOWMEIM v1.0, es realizar la mejora de la imagen de la manera requerida por el usuario, procesa la imagen, extrae rasgos. SOWMEIM v1.0 fue desarrollado en MATLAB, permite a los usuarios capturar e invocar imágenes disponibles en el disco duro. Una vez efectuado el procesamiento y por ende la mejora de la imagen, el usuario podrá guardar su imagen resultado en los formatos que soporta MATLAB. También, en futuro se proyecta como una herramienta de Visión por Computadora total, esto gracias a las múltiples aplicaciones que tiene esta ciencia; en ingeniería, industria, diseño gráfico, seguridad, medicina, geología, biología, química, etc. Software similares han sido desarrollados para mejorar las propiedades y mostrar vistas particulares de las imágenes al usuario; a través del mundo, tal es el caso del: Adobe Photoshop, Image Tool Application, Area Sat, entre muchos otros; pero claro estos programas comerciales, ocupan mucho espacio de memoria y muchas veces muy lentos en sus funciones. Con respecto a Cotopaxi y nuestra universidad creemos que el SOWMEIM v1.0 es uno de los primeros, y esperamos no sea el único en el área de Inteligencia Artificial.

Múltiples actividades tanto investigativas, en complicidad con los conocimientos adquiridos durante el estudio de esta carrera fueron de imperante necesidad en el transcurso del desarrollo del SOWMEIM. El primer punto desarrollado es la búsqueda de un problema y por ende, una solución eficiente al mismo. Para ello planteamos lo siguiente:

Objetivo general:

Implementar un software prototipo de procesamiento y extracción de rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial, encaminaban a la creación de un software con su primera versión capacitada para

mejorar la visualización de una imagen tratada por un estudiante de la especialización de Diseño Gráfico, que en este caso es el usuario final.

Objetivos específicos:

Concluir con un software prototipo de excelentes características.

Una interfaz adecuada y fácil de usar.

Optimizar en la construcción de algoritmos de procesamiento de imágenes.

Mostrar los rasgos de la imagen de una manera primordial.

Elaborar de un manual de usuario para el correcto manejo de SOWMEIM v1.0.

Hipótesis:

La implementación de un software prototipo de procesamiento y extracción de rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial, ayudará a la especialización de Diseño Gráfico de C.I.Y.A. en la Universidad Técnica de Cotopaxi.

Esto enmarca la utilidad del SOWMEIM v1.0 para ayudar a la especialización de Diseño Gráfico. Se une a esto los diferentes tipos de investigación utilizados como son: La investigación aplicada, la investigación bibliográfica, la investigación de campo, y la investigación experimental; todas ellas necesarias en cada aspecto y detalle de un proyecto investigativo. Métodos investigativos como el científico, deductivo, e inductivo – deductivo; son los conductores que hacen posible llevar un investigación no errónea en cualquier tipo de proyecto científico – técnico.

La población participe para esta investigación y la que desarrollo la encuesta utilizada para la adquisición relevante de información promotora del software prototipo SOWMEIM v1.0; estaba planteada de 102 personas, pero en la práctica fue de 100 personas exactamente, distribuidas entre estudiantes de cuarto, sexto, séptimo, octavo y los respectivos docentes de la especialización de Diseño Gráfico. Dejando como resultado final en las nueve preguntas el mayor porcentaje

para la respectiva apertura y necesidad de la implementación del software prototipo SOWMEIM v1.0.

Ya para concluir el proyecto investigativo denominado SOWMEIM v1.0 nos enfocamos en las tareas desarrolladas y los aportes con esta investigación, como:

Procesamientos básicos con ciertas imágenes.

Extracción de rasgos en imágenes.

Mejorar la percepción visual de las imágenes de práctica.

Hacer de nuestra aplicación una herramienta que se pueda instalar en cualquier computador que cumpla con los requerimientos necesarios.

Y con respecto a los aportes es claro entender que el proyecto SOWMEIM v1.0 abre el escenario en el tema de Visión Artificial, además de ser una herramienta útil para los estudiantes de Diseño Gráfico, mismos que podrán trabajar con sus imágenes en un cierto grado de ventaja con respecto a otras de la misma función: mejorar la imagen, extraer los rasgos, obtener un segmento, cambiar de formato, cambiar de percepción, entre otras cosa. También cabe mencionar que sirve de gran ayuda para los estudiantes de Ingeniería en Informática y Sistemas Computacionales que se inclinen por este tipo de temas investigativos.

La tesis se organiza de la siguiente manera: en el *capítulo uno* se cita la información teórica de todo lo inherente sobre: el ojo humano, el sistema de visión humano, MATLAB, conceptos, requerimientos, procesamiento de imágenes, métodos, técnicas, etc. El *capítulo dos* describe los análisis de resultados de las encuestas hechas a los estudiantes y docentes de la especialización en Diseño Gráfico, Descripción de esta especialización, importancia de las imágenes en diseño gráfico, aplicaciones en general del procesamiento de imágenes, esquemas de un sistema de Visión por Computadora, Metodología, entre otras cosas irrelevantes. El *capítulo tres* establece el diseño, desarrollo, figuras, y todo lo referente a la implementación del software prototipo SOWMEIM v1.0, manual del usuario e información adicional. Al final de este capítulo están presentes las conclusiones del trabajo, recomendaciones, bibliografía y anexos.

“Visión es el arte de ver las cosas invisibles”

Los viajes de Gulliver (1726)

Jonathan Swift.

CAPITULO I

“EL SER HUMANO Y LA VISIÓN POR COMPUTADORA”

1.1 Introducción

La Inteligencia Artificial es la ciencia que, mediante un conjunto de técnicas o herramientas, pretende modelar un comportamiento, obtener una serie de pautas de actuación o extraer un conjunto de características a partir de un problema objeto, es decir, es el campo de estudio que se enfoca a la explicación y emulación de la conducta inteligente en función de procesos computacionales. Como un campo de aplicación de la Inteligencia Artificial, el PDI (Procesamiento Digital de Imágenes) encuentra su base en muy distintas técnicas, cada una de ellas orientada a resolver cierta dificultad relacionada con la propia imagen digital.

Según el grupo investigador apoyados en las ideas antes descritas, mencionamos que han pasado más de dos décadas desde que varios investigadores y grupos de investigación comenzaron a intentar automatizar el proceso de visión humana, y sin embargo prevalece un cierto sentido de frustración. Esta frustración existe debida, entre otras cosas, a la propia sofisticación del sistema de percepción humana.

González R. y Wood R en su libro “Digital Image Processing” afirma que el PDI ha crecido considerablemente durante la última década con la utilización aumentada de la Ingeniería de Imágenes. Además ha encontrado un papel significativo en las aplicaciones científicas, industriales, espaciales, y gubernamentales.

El PDI es un término genérico para la manipulación de imágenes con objeto de producir nuevas imágenes que son mejores, en algún sentido. El PDI comprende la mejora de imágenes, que modifica una imagen para mejorar su apariencia a nivel humano: la restauración de imágenes que corrige la degradación de imágenes, que pretende la representación de imágenes, de forma compacta manteniendo un cierto nivel de calidad.

Según el grupo investigador esta tesis enfoca la mejora o adaptación de la imagen; por medio de un Software prototipo y es claro mencionar la concordancia que tiene estos autores del libro con nuestro criterio.

Lo que resta del presente capítulo se dedica a definir: el sistema de visión humano con su componente principal, el ojo; el sistema de visión por computadora con sus patrocinadores, el computador y la cámara digital; los fundamentos básicos de operación en el PDI que se utilizarán en el resto del trabajo. Los Apartados del 1.4 al 1.6 definen el origen, las propiedades básicas, comandos, funciones, requerimientos, y otras características importantes de la herramienta MATLAB. El Apartado 1.7 define la formación de la imagen, transformación de Fourier, histogramas y el PDI. Por último el Apartado 1.8 define la extracción de rasgos y el Apartado 1.9 menciona introducción a la descripción de la textura y segmentación.

1.2 El Sistema de Visión Humano

Según la página de Internet <http://www.monografías.com/trabajos3/visionh> dice que, La visión humana es un sistema sofisticado que se da cuenta de actos en los estímulos visuales. Este tiene una evolución por millones de años, principalmente

para defensa o supervivencia. Intuitivamente, la computadora y la visión humana parecen tener la misma función. El propósito de ambos sistemas es interpretar los datos espaciales, datos que se ponen en un índice por más de una dimensión. Aunque la computadora y la visión humana son funcionalmente similares, usted no puede esperar un sistema de visión por computadora para reproducir la función del ojo humano exactamente. Esto es en parte porque no se entiende totalmente cómo el ojo trabaja.

En la visión humana, el elemento de sentido es el ojo; del cual, se transmiten las imágenes vía del nervio óptico al cerebro, para procesarlas. El nervio óptico tiene la insuficiente capacidad de llevar toda la información captada por el ojo. De acuerdo con esto, debe haber algún pre-procesamiento de la imagen antes que se transmita al nervio óptico. El sistema de visión humano puede ser moldeado en tres partes: El ojo, El sistema neural, y el Procesamiento por el cerebro.

1.2.1 El Ojo

El ojo humano es un sistema óptico formado por un dioptrio esférico y una lente, que reciben, respectivamente, el nombre de córnea y cristalino, y que son capaces de formar una imagen de los objetos sobre la superficie interna del ojo, en una zona denominada retina, que es sensible a la luz.

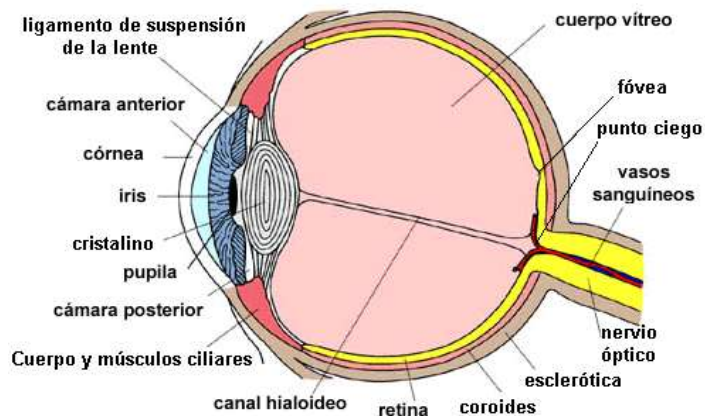


Figura 1.0: El Ojo Humano

En la Figura 1.0 se observa las partes que forman el ojo. Tiene forma aproximadamente esférica y está rodeado por una membrana llamada **esclerótica** que por la parte anterior se hace transparente para formar la **córnea**.

Tras la córnea hay un diafragma, el **iris**, que posee una abertura, la **pupila**, por la que pasa la luz hacia el interior del ojo. El iris es el que define el color de nuestros ojos y el que controla automáticamente el diámetro de la pupila para regular la intensidad luminosa que recibe el ojo.

El **crystalino** está unido por ligamentos al **músculo ciliar**. De esta manera el ojo queda dividido en dos partes: la posterior que contiene **humor vítreo** y la anterior que contiene **humor acuoso**. El cristalino enfoca las imágenes sobre la envoltura interna del ojo, la **retina**. Esta envoltura contiene fibras nerviosas (prolongaciones del nervio óptico) que terminan en unas pequeñas estructuras denominadas conos y bastones muy sensibles a la luz. Existe un punto en la retina, llamado **fóvea**, alrededor del cual hay una zona que sólo tiene conos (para ver el color). Durante el día la fóvea es la parte más sensible de la retina y sobre ella se forma la imagen del objeto que miramos.

Los millones de nervios que van al cerebro se combinan para formar un **nervio óptico** que sale de la retina por un punto que no contiene células receptoras. Es el llamado **punto ciego**.

La córnea refracta los rayos luminosos y el cristalino actúa como ajuste para enfocar objetos situados a diferentes distancias. De esto se encargan los músculos ciliares que modifican la curvatura de la lente y cambian su potencia. Para enfocar un objeto que está próximo, es decir, para que la imagen se forme en la retina, los músculos ciliares se contraen, y el grosor del cristalino aumenta, acortando la distancia focal imagen. Por el contrario si el objeto está distante los músculos ciliares se relajan y la lente adelgaza. Este ajuste se denomina **acomodación o adaptación**.

Según el grupo investigador podemos resumir que el **SISTEMA DE VISIÓN HUMANO** es un conjunto de procesos en los que una persona capta los objetos de su realidad a través de los ojos y con vinculación al cerebro permite generar una imagen de esos objetos en una parte del ojo llamada retina.

1.3 Sistema de Visión de Computadora

Según la página de Internet <http://www.monografías.com/trabajos4/visionco> dice que, dado el progreso en la tecnología de la computadora, el hardware de visión por computadora es ahora relativamente barato; un sistema de visión por computadora básico requiere una cámara, una interfaz de la cámara y una computadora. En la actualidad, algunas computadoras personales ofrecen la capacidad para un sistema de visión básico, incluyendo una cámara y su interfaz dentro del sistema. Allí se especializa los sistemas para la visión, ofreciendo alta actuación en más de un aspecto; pero paulatinamente, estos pueden ser caros, cuando se trata de un sistema especialista.

1.3.1 Cámaras

Según la página de Internet <http://electronica.udea.edu.co/cursos/poli.htm> dice que, Una cámara es un dispositivo que captura una imagen del mundo real, para convertir la información contenida de dicha imagen a signos electrónicos correspondientes a su información que pueden guardarse de una manera reproducible. En una cámara convencional la información de la imagen es guardada en un rollo a manera de signos químicos para posteriormente ser revelados manualmente. Al contrario de una cámara electrónica, que convierte la información de la imagen captada a signos electrónicos para ser almacenados en una memoria. Definición que nos parece la más importante acerca de las cámaras, en especial la definición de una cámara digital.

Según el grupo investigador la clasificación que da esta página de Internet a las cámaras, es sin duda la ideal ya que como es conocimiento de todos, hace pocos años para retratarnos o retratar algo en una foto era indispensable la adquisición de rollos y hoy en la actualidad el tomar una foto depende del dinero que se tenga para conseguir en el mercado un cámara de exquisitas características.

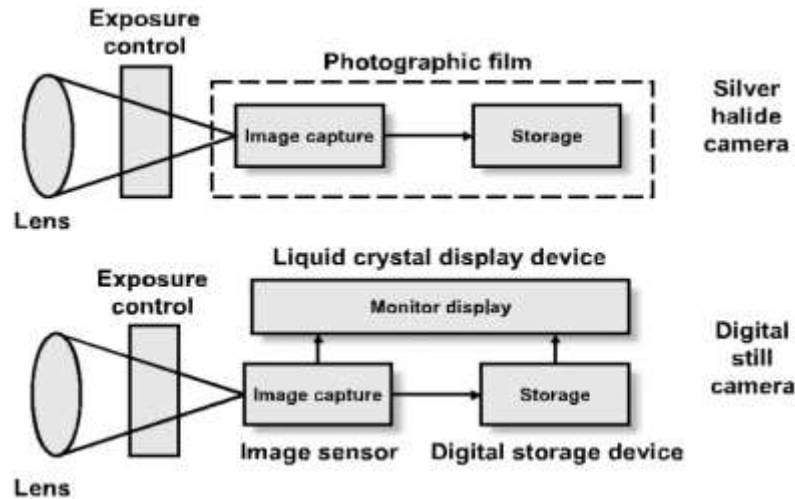


Figura 2.0: Diferencia entre una cámara convencional y una cámara digital

Las cámaras son divididas en dos grupos según el método de captura de la imagen: las cámaras convencionales y las cámaras electrónicas. Las cámaras electrónicas son nuevamente divididas en dos grupos: analógicas y digitales.

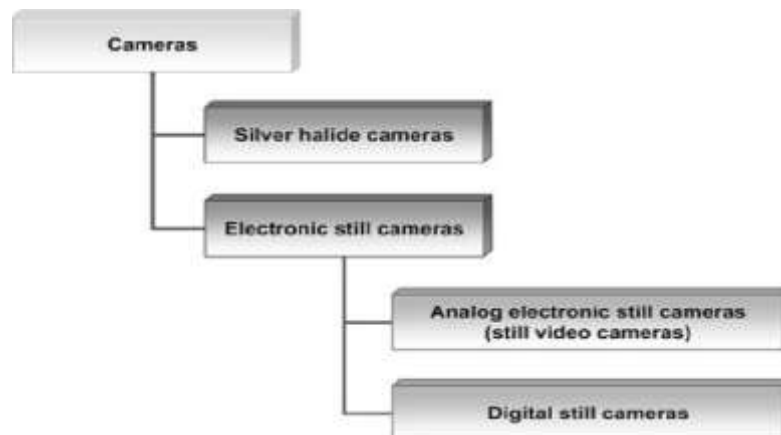


Figura 2.1: Clasificación de las Cámaras

Cámara Digital es el dispositivo usado en el trabajo con gráficos por computadora a través de un interfaz. Este dispositivo emplea su funcionalidad para capturar imágenes reales y digitalizarlas, para posteriormente almacenarlas en memoria en forma de mapas de bits, con la ayuda de una placa de circuito especial.

1.3.2 Interfaces de Computadora

En el libro de Nixon M. y Aguado A. “Feature extraction and Image Processing”, en la pág. 97 dice que; la interfaz de la computadora básica necesita convertir un signo analógico de una cámara en un juego de números digitales. El sistema de la interfaz se llama framegrabber (grabador de escena), desde que este sujeta los datos de la imagen de una secuencia de video (Figura 3.0). Note que ciertas cámaras inteligentes que proporcionan la información digital no necesitan esta interfaz particular, ya que permite el almacenamiento inmediato de sus datos. Sin embargo, la señal de la cámara analógica es continua y se transforma en digital usando un formato conversor de análogo a digital. Normalmente se usan los conversores Flash que requieren alta velocidad para la conversión (es decir 11MHz que no puede compararse por cualquier otra tecnología de conversión). La señal de video requiere una conversión de prioritarias condiciones; esto incluye la restauración de DC para asegurar que el nivel de DC correcto se atribuye a la señal de video entrante. Usualmente, son usados los conversores de 8-bit A/D a 6 dB/bit, esto da 48 dB. El rendimiento del conversor de A/D se alimenta a menudo del look-up table (LUTs) que implementa la conversión de los datos de entrada, pero en el hardware, en lugar de en el software, de una manera muy rápida. Entonces se guardan las salidas del conversor de A/D en la memoria de la computadora. Esto ahora se realiza por el puerto-dual de memoria, que es compartido por la computadora y el grabador de escena. El grabador de escena sólo toma mando de la memoria de la imagen cuando está adquirida, y guardada.

Según el grupo investigador la teoría que proporcionan Nixon y Aguado en su libro es muy eficiente para el entendimiento en lo que se refiere a interfaces de computadora; que en otras palabras no es más que el entorno que muestra la computadora al usuario de un determinado software. Y en el caso de esta tesis por la utilización de una cámara digital tendremos en parte conocimientos del interfaz de la misma.

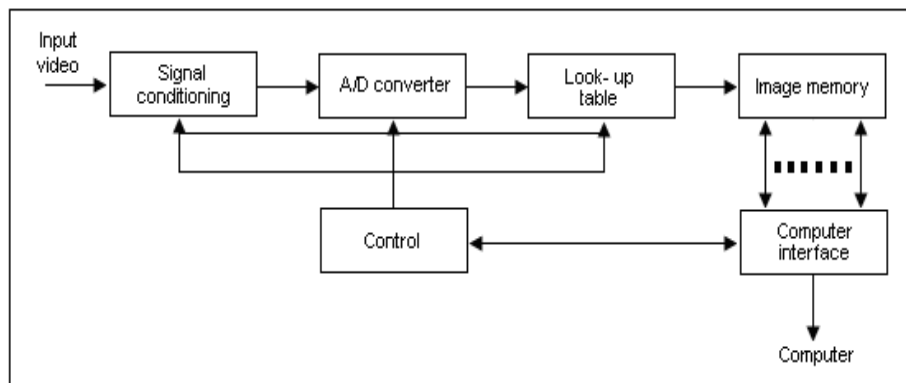


Figura 3.0: Interfaz de computadora

1.3.3 Procesando una Imagen

El Libro de los autores Cuevas E. y Zaldivar D. “Visión por Computador utilizando MATLAB y el Toolbox de Procesamiento Digital de Imágenes” en las pág. (18-29) nos ayuda a definir y congeniar en los temas de procesamiento de imágenes y entre otras cosas y en general que el procesamiento de imágenes y las técnicas de visión por computadora se llevan a cabo en el software de la computadora. A menudo, solo las técnicas más simples emigran al hardware; aunque codificando las técnicas para aumentar al máximo la eficacia en la transmisión de la imagen son de interés comercial, y el desarrollo del hardware, muy sofisticado. Los sistemas incluyen códigos de formatos de imagen Joint Photographic Expert Group (JPEG) (Unión de Grupos Especialistas Fotográficos) y el Moving Picture Expert Group (MPEG) (Movimiento de Grupos Expertos en Fotografía). C y C++ son ahora los lenguajes más populares para la aplicación de un sistema de visión. C debido a sus estupendas funciones integradas alto y bajo

nivel, y la disponibilidad de compiladores buenos. Como sistemas más complejos. C++ se pone más atractivo cuando pueden explotarse encapsulamientos y polimorfismos. Muchas personas usan Java ahora como un lenguaje de desarrollo, en parte debido a la independencia de su plataforma, pero también debido a la facilidad en la aplicación (aunque alguna demanda de velocidad/eficiencia no están buena como en C/C++). Hay una ventaja considerable de aplicación asociada con el uso Java™ Advanced Imaging API (Application Programming Interface) (Imagen Avanzada) (Interfaz de aplicaciones programadas). Hay algunos libros de texto que ofrecen sistemas de procesamiento de imágenes implementados en estos lenguajes. Hay también, muchos paquetes comerciales disponibles aunque éstos se limitan a menudo a las técnicas básicas y no incluye las técnicas más sofisticadas de extracción de forma.

Según el grupo investigador el sistema de procesamiento de imágenes de Khoros ha atraído mucho interés; este, es esquemático, dónde un usuario se une junto con los módulos escogidos. Esto permite una buena visualización del flujo de información durante el procesamiento. Sin embargo, la matemática subrayada no es de hecho clara para el usuario, como puede ser cuando se usa un sistema matemático.

1.4 Herramientas Matemáticas

El libro de Nixon M. y Aguado A. “Feature extraction and Image Processing”, en la pág.128 dice que; en los recientes años se han desarrollado varios sistemas matemáticos. Éstos ofrecen lo que es virtualmente un sistema procesador de palabras para matemáticos y muchos son basados usando la pantalla de un sistema de Windows. La ventaja de estos sistemas es que usted puede, bastante bien, transponer directamente la matemática de los libros de texto, y ve cómo funciona. La funcionalidad del código no se disimula por el uso de estructuras de datos, aunque esto puede hacer representar al código, embarazoso. Una ventaja mayor es que el sistema proporciona una funcionalidad de bajo nivel y una visualización de

esquemas de datos, permitiéndole al usuario concentrarse exclusivamente en las técnicas.

Según el grupo investigador de acuerdo con estos sistemas se permiten el lujo de dar una ruta excelente para tener una apreciación prioritaria al desarrollo del código de la aplicación de los sistemas matemáticos y verificar el correcto funcionamiento del código final.

Mathcad, Mathematica, Maple y MATLAB son las herramientas matemáticas actuales, más populares. Ha habido estudios que comparan su eficacia, pero es difícil de asegurar la comparación precisa debido a la velocidad impresionante de técnicas de desarrollo. La mayoría de los sistemas tiene sus protagonistas y detractores, como en cualquier sistema comercial. Hay muchos libros que usan estos paquetes para asuntos particulares, y hay a menudo manuales como aditamentos a los paquetes. Por lo que usaremos MATLAB en el trayecto de la tesis, que junto con Mathcad son quizás los dos más ventajosos de los sistemas matemáticos. A continuación describiremos en el siguiente apartado la herramienta MATLAB.

1.5 MATLAB

Según la página de Internet <http://www.monografías.com/matlab.html> dice que, MATLAB es un entorno de computación y desarrollo de aplicaciones totalmente integrado orientado para llevar a cabo proyectos en donde se encuentren implicados elevados cálculos matemáticos y la visualización gráfica de los mismos. MATLAB integra análisis numérico, cálculo matricial, proceso de señal y visualización gráfica en un entorno completo donde los problemas y sus soluciones son expresados del mismo modo en que se escribirían tradicionalmente, sin necesidad de hacer uso de la programación tradicional.

Según el grupo investigador se le acoge como una excelente definición que sin duda alguna globaliza lo que hace y papel fundamental que cumple MATLAB en

un computador y lo que permite que realice el usuario para sus necesidades o problemas, además de todo la estructura de MATLAB brinda a los Informáticos la facilidad de utilizar demos, simuladores, realidad virtual, etc.

1.5.1 Origen de MATLAB

Según la página electrónica <http://www.mathworks.com> dice que, MATLAB fue originalmente desarrollado en lenguaje FORTRAN para ser usado en computadoras mainframe. Al pasar de los años fue complementado y reimplementado en lenguaje C. Actualmente la licencia de MATLAB es propiedad de MathWorks Inc. El nombre de MATLAB proviene de la contracción de los términos **MAT**rix **LAB**oratory y fue inicialmente concebido para proporcionar fácil acceso a las librerías LINPACK y EISPACK, las cuales representan hoy en día dos de las librerías más importantes en computación y cálculo matricial.

1.5.2 ¿Qué es MATLAB?

Según la página electrónica <http://www.mathworks.com> dice que, MATLAB es un programa para realizar cálculos numéricos con **vectores** y **matrices**. Como caso particular puede también trabajar con números escalares, tanto reales como complejos, con cadenas de caracteres y con otras estructuras de información más complejas. Una de las capacidades más atractivas es la de realizar una amplia variedad de **gráficos** en dos y tres dimensiones. Además posee un lenguaje de programación propio. Es decir, MATLAB es un gran software de cálculo técnico y científico. Para ciertas operaciones es muy rápido, cuando puede ejecutar sus funciones en código nativo con los tamaños más adecuados para aprovechar sus capacidades de vectorización. En otras aplicaciones resulta bastante más lento que el código equivalente, desarrollado en C/C++ o Fortran.

1.5.3 Plataformas, Requerimientos y Estructura de MATLAB

Plataformas

Según la página electrónica <http://www.mathworks.com/capable> dice que, MATLAB está disponible para un amplio número de plataformas: estaciones de trabajo SUN, Apollo, VAXstation y HP, VAX, MicroVAX, Gould, Apple Macintosh y PC AT compatibles 80386 o superiores. Opera bajo sistemas operativos UNIX, Macintosh y Windows.

Requerimiento de sistema en Windows

	Operating System	Processors	Disk Space	RAM
32-bit MathWorks Products	Windows XP <ul style="list-style-type: none"> ▪ Service Pack 1 or 2 	Pentium III Pentium IV Pentium Xeon Pentium M	460 MB (MATLAB only)*	512 MB 1024 MB (recommended)
	Windows 2000 <ul style="list-style-type: none"> ▪ Service Pack 3 or 4 Windows Server 2003	AMD Athlon AMD Athlon MP AMD Athlon XP AMD Athlon 64 AMD Opteron		
64-bit MathWorks Products	Windows XP x64	Intel EM64T processors AMD64 processors	460 MB (MATLAB only)*	512 MB 1024 MB (recommended)

Requerimiento de sistema en Linux

	Operating System	Processors	Disk Space	RAM
32-bit MathWorks Products	Linux – built using <ul style="list-style-type: none"> ▪ kernel: 2.4.x or 2.6.x ▪ glibc: 2.3.2 & above 	Pentium III Pentium IV Pentium Xeon Pentium M AMD Athlon AMD Athlon MP AMD Athlon XP AMD Athlon 64 AMD Opteron	460 MB (MATLAB only)	512 MB 1024 MB (recommended)
64-bit MathWorks Products	Linux – built using <ul style="list-style-type: none"> ▪ kernel: 2.4.x or 2.6.x ▪ glibc: 2.3.4 & above 	Intel EM64T processors AMD64 processors	460 MB (MATLAB only)	512 MB 1024 MB (recommended)

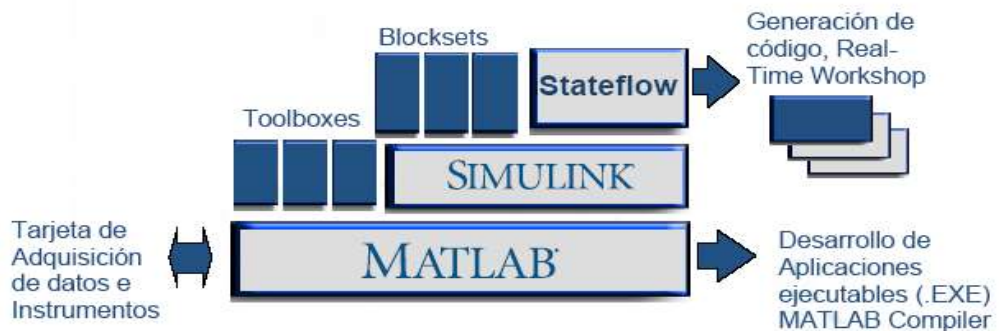


Figura 4.0: Estructura de MATLAB.

Según el grupo investigador los conceptos que nos han proporcionado la página de Internet de la compañía dueña de los derechos y licencias de MATLAB como es MathWorks, cuya semántica dejan en claro muchas ideas infaltables en nuestra cultura como estudiantes de la ingeniería informática. Podemos constatar nuestros conocimientos al trabajar por primera vez en el entorno MATLAB que verdaderamente es un lenguaje que hoy en día se ha convertido en fundamental a tratar en las clases de carreras técnicas de ingeniería o tecnología. Los requerimientos también están dotados a la tecnología de hardware actual de la mano con la posibilidad de ensamblaje de una computadora. Es decir al alcance de una computadora sin costos en su hardware elevados.

1.5.4 Utilidades y Funcionalidades

Utilidades

Según el grupo investigador mencionamos algunas de las múltiples utilidades y funcionalidades que proporciona el programa MATLAB:

- Gestión y mantenimiento de errores.
- Álgebra lineal.
- Funciones matemáticas elementales y especializadas.
- Operadores lógicos y aritméticos.
- Matrices elementales y manipulación de vectores.
- Matrices especiales.
- Estadística básica y análisis de datos.
- Polinomios e interpolación.
- Entradas y Salidas.
- Conversión de tipos de datos Fortran.

Funcionalidades

- Adquisición de Datos.
- Exploración y análisis de datos.
- Análisis numérico y simbólico.
- Visualización, modelación y simulación avanzada.

- Procesamiento de señales e imágenes.
- Desarrollo de algoritmos.
- Programación y desarrollo de aplicaciones.

1.6 Algunos Comandos y Herramientas de MATLAB

Según la ayuda del programa MATLAB obtenemos los siguiente:

clc.- Despeja la ventana de comandos.

clear.- No afecta a las ventanas, pero si borra todas las variables de la memoria.

clf.- Borra la figura actual y por tanto despeja la ventana de gráficos.

demo.- Inicia el MATLAB EXPO, un entorno gráfico de demostración que ilustra algunos tipos de operaciones que se pueden realizar con MATLAB.

exit.- Sirve para salir de MATLAB.

help.- Sirve para abrir la ayuda de MATLAB.

MATLAB también ofrece una familia de aplicaciones que se han dado en llamar Cajas de Herramientas o "Toolboxes". Un aspecto muy importante para la mayoría de los usuarios de MATLAB es que las toolboxes son colecciones de funciones, códigos o macros de MATLAB, conocidos como M-files, muy fáciles de comprender y de usar, que extienden el ambiente del programa con el fin de resolver clases particulares de problemas.

Según el grupo investigador en algunas de las áreas para las cuales existen toolboxes son: Procesamiento de Señales, Análisis y Diseño de Sistemas de Control, Simulación Dinámica de Sistemas, Identificación de Sistemas, Redes Neuronales, Ecuaciones Diferenciales Parciales, Procesamiento de Imágenes, Matemática Simbólica, Lógica Difusa, Estadística, Finanzas, etc.

1.6.1 Manipulación de Vectores y Matrices

Con la ayuda del libro de Guerrero Diego “Apuntes de MATLAB” y el manual “Aprenda MATLAB 7.0 como si estuviera en primero” de la Universidad Politécnica de Madrid, podemos llegar a la conclusión en muchas líneas teóricas necesarias para el entendimiento de la presente tesis y en si de la herramienta MATLA. Lo que hace verdaderamente poderoso a MATLAB es la facilidad para realizar operaciones matemáticas con elementos como: **Vectores y Matrices**. Los vectores y matrices en MATLAB se trabajan igual en cuanto a asignación, por eso se explican juntos. Pero las operaciones posibles, si son diferentes.

Si se tienen dos matrices (o vector y matriz, o dos vectores), y se quieren: sumar, multiplicar ó restar sólo es necesario anotar esta operación normalmente (como se haría con números). Entre otras operaciones elementales tenemos:

- Sumar y Restar Matrices
- Multiplicar Matrices
- Producto escalar
- Producto de una matriz por un vector
- Dividiendo Matrices
- Usar Exponentes con Matrices
- Funciones Matriciales Trascendentales y Elementales.

1.7 Imágenes, Probando y Procesando

1.7.1 Introducción

El Libro de los autores Cuevas E. y Zaldivar D. “Visión por Computador utilizando MATLAB y el Toolbox de Procesamiento Digital de Imágenes” en conjunto con el Image Processing ToolBox versión 6.0. User’s Guide abordan los conocimientos necesarios para el entendimiento ideal de cómo se lleva a cabo el procesamiento de imágenes en MATLAB además de dotar de información

necesaria para explotar en la herramienta por nosotros. Y es sí dejando en claro que la implementación de algoritmos en visión por computador resulta muy costoso en tiempo, ya que se requiere de la manipulación de punteros, gestión de memoria, etc. Hacerlo en lenguaje C++ (que por sus características compartidas de alto y bajo nivel lo hacen el más apropiado para la implementación de algoritmos de visión computacional) supondría la inversión de tiempo y sin la seguridad de que lo que queremos implementar funcionará. Además utilizar C++ para el periodo de prueba exige un tiempo normal de corrección de errores debidos al proceso de implementación del algoritmo, es decir errores programáticos efectuados por ejemplo al momento de multiplicar dos matrices, etc. Todos estos problemas pueden ser resueltos si la implementación de prueba es realizada en MATLAB utilizando su toolbox de procesamiento de imágenes con ello el tiempo de implementación se convierte en el mínimo con la confianza de utilizar algoritmos científicamente probados y robustos.

El toolbox de procesamiento de imágenes contiene un conjunto de funciones de los algoritmos más conocidos para trabajar con imágenes binarias, transformaciones geométricas, morfología y manipulación de color que junto con las funciones ya integradas en MATLAB permite realizar análisis y transformaciones de imágenes en el dominio de la frecuencia (transformada de Fourier y Wavlets).

Según el grupo investigador a partir de las siguientes líneas teóricas podemos mencionar, en su totalidad que sistemáticamente van indicando los procesos y pasos que se pueden tomar para realizar un procesamiento de imágenes según la necesidad del usuario.

1.7.1.1 Formación de la Imagen

En MATLAB una imagen a escala de grises es representada por medio de una matriz bidimensional de $m \times n$ elementos en donde n representa el numero de píxeles de ancho y m el numero de píxeles de largo. El elemento v_{11} corresponde

al elemento de la esquina superior izquierda (ver Figura 5.0), donde cada elemento de la matriz de la imagen tiene un valor de 0 (negro) a 255 (blanco). Por otro lado una imagen de color RGB (la más usada para la visión computacional, además de ser para MATLAB la opción default) es representada por una matriz tridimensional $m \times n \times p$, donde m y n tienen la misma significación que para el caso de las imágenes de escala de grises mientras p representa el plano, que para RGB que puede ser 1 para el rojo, 2 para el verde y 3 para el azul. La Figura 5.1 muestra detalles de estos conceptos.

Los formatos de imágenes que soporta MATLAB son mostrados en la siguiente tabla. Para introducir una imagen guardada en un archivo con alguno de los formatos especificados en el cuadro, solo tiene que usarse la función *imread* y asignar su resultado a una variable que representará a la imagen (de acuerdo a la estructura, Figura 5.0 para representar escala de grises y Figura 5.1 para RGB).

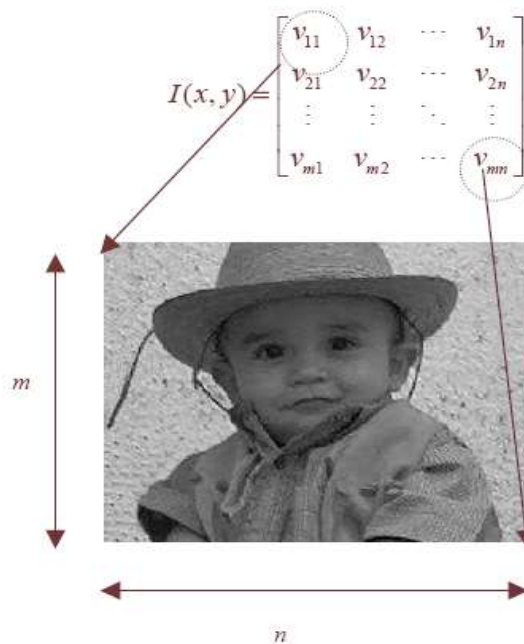


Figura 5.0: Representación de una imagen a escala de grises en MATLAB.

Formato	Extensión
TIFF	.tiff
JPEG	.jpg

GIF	.gif
BMP	.bmp
PNG	.png
XWD	.xwd

Cuadro N° 1.0: Formatos y extensiones soportadas por MATLAB.

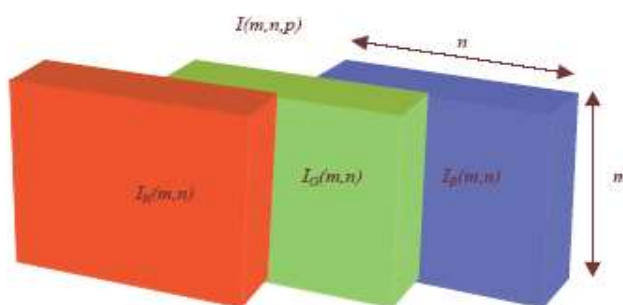
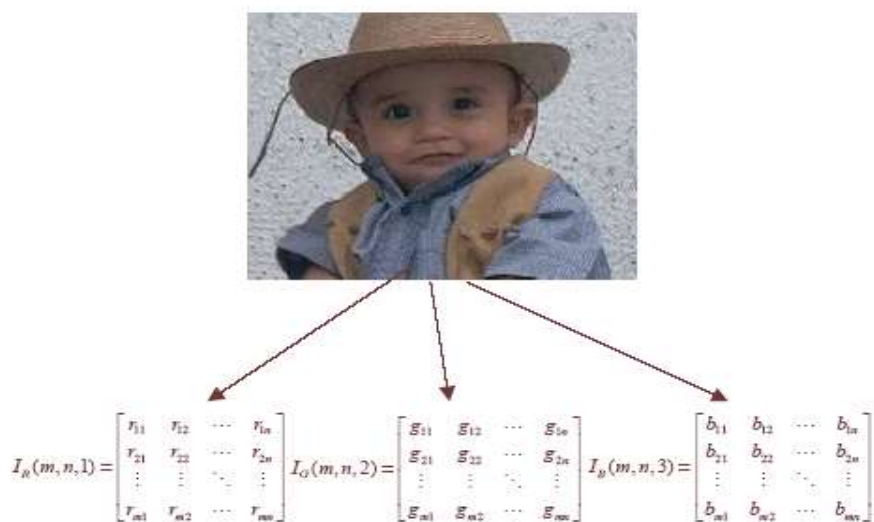


Figura 5.1: Representación de una imagen a color RGB en MATLAB.

1.7.1.2 Píxel y Planos en las Imágenes

El acceso a píxel de una imagen es una de las operaciones más comunes en visión computacional y en MATLAB está sumamente simplificado; solo bastará con

indexar el píxel de interés en la estructura de la imagen. Consideremos que tenemos una imagen *image1* en escala de grises y deseamos obtener su valor de intensidad en el píxel especificado por $m=100$ y $n=100$.

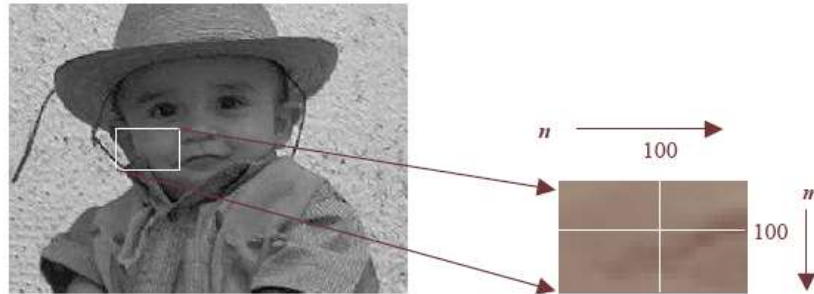


Figura 5.2: Obtención del valor de un píxel

En el caso de imágenes a escala de grises estas solo tienen un plano, constituido por la matriz $m \times n$ que contiene los valores de intensidad para cada índice. Sin embargo las imágenes de color cuentan con más de un plano. En el caso de imágenes RGB estas cuentan con 3 planos uno para cada color que representa.

Los planos resultantes, por ejemplo, son mostrados en la Figura 5.3.

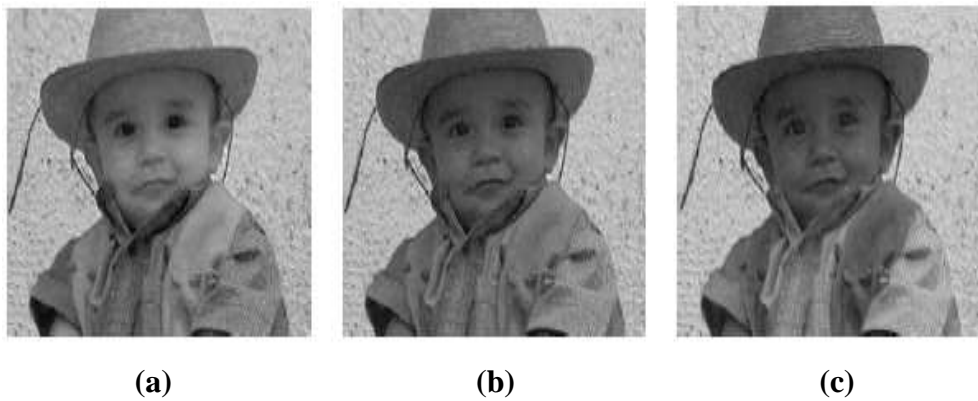


Figura 5.3: Planos de la imagen a) rojo, b) verde y c) azul.

1.7.1.3 Sub-Muestreo de Imágenes

En ocasiones es necesario hacer cálculos que requieren procesar por completo la imagen, en estos casos hacerlo sobre la resolución original de la imagen sería muy costoso. Una alternativa más eficiente, resulta el sub-muestreo de la imagen. Sub-muestreo significa generar una imagen a partir de tomar muestras periódicas de la imagen original, de tal forma que esta quede más pequeña. Si se considera la imagen $I(m,n)$ definida como:

$$I(m, n) = \begin{bmatrix} v_{11} & v_{12} & v_{13} & \cdots & v_{1n} \\ v_{21} & v_{22} & v_{23} & \cdots & v_{2n} \\ v_{31} & v_{32} & v_{33} & \cdots & v_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & v_{m3} & \cdots & v_{mn} \end{bmatrix}$$

Y se desea sub-muestrear la imagen para obtener la mitad de su tamaño original, así la nueva imagen quedaría compuesta por los elementos tomando *uno si y otro no* de la imagen original:

$$I_{S_2}(m, n) = \begin{bmatrix} v_{11} & v_{13} & v_{15} & \cdots & v_{1(n-2)} \\ v_{31} & v_{33} & v_{35} & \cdots & v_{3(n-2)} \\ v_{51} & v_{53} & v_{55} & \cdots & v_{5(n-2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{(m-2)1} & v_{(m-2)3} & v_{(m-2)5} & \cdots & v_{(m-2)(n-2)} \end{bmatrix}$$

El mismo concepto se explica gráficamente en la Figura 5.4.

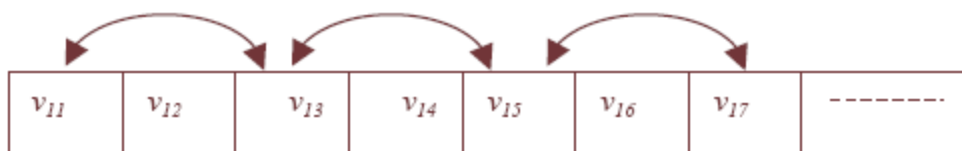


Figura 5.4: Ejemplo de sub-muestreo

1.7.1.4 *Procesamiento de la Imagen*

El número de funciones que implementa el toolbox para el procesamiento de imagen es muy diverso, sin contar la múltiple oferta de funciones ya generada por otros usuarios y disponibles a través del Internet, sin embargo trabajaremos con algunas consideradas como las más usadas y útiles para la visión computacional.

1.7.1.5 *Filtraje Espacial*

Es la operación que se aplica a una imagen para resaltar o atenuar detalles espaciales con el fin de mejorar la interpretación visual o facilitar un procesamiento posterior, y constituye una de las técnicas comprendidas dentro del realce de imágenes. El filtrado espacial es una operación "local" en procesamiento de imagen, en el sentido de que modifica el valor de cada píxel de acuerdo con los valores de los píxeles que lo rodean; se trata de transformar los niveles de gris originales de tal forma que se parezcan o diferencien más de los correspondientes a los píxeles cercanos.

El filtraje espacial es utilizado, para realizar efectos de eliminación de ruido o bien detección de bordes. En ambos casos la determinación de los píxeles de la nueva imagen depende del píxel de la imagen original y sus vecinos. De esta forma es necesario configurar una matriz (mascara o ventana) que considere cuales vecinos y en qué forma influirán en la determinación de el nuevo píxel. Consideremos una imagen $IS(m; n)$.

$$I_S(m, n) = \begin{bmatrix} v_{S11} & v_{S12} & v_{S13} & \dots & v_{S1n} \\ v_{S21} & v_{S22} & v_{S23} & \dots & v_{S2n} \\ v_{S31} & v_{S32} & v_{S33} & \dots & v_{S3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{Sm1} & v_{Sm2} & v_{Sm3} & \dots & v_{Snn} \end{bmatrix}$$

Que será la imagen a la cual pretendemos filtrar espacialmente y $IT(m; n)$:

$$I_T(m, n) = \begin{bmatrix} vt_{11} & vt_{12} & vt_{13} & \cdots & vt_{1n} \\ vt_{21} & vt_{22} & vt_{23} & \cdots & vt_{2n} \\ vt_{31} & vt_{32} & vt_{33} & \cdots & vt_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ vt_{m1} & vt_{m2} & vt_{m3} & \cdots & vt_{mn} \end{bmatrix}$$

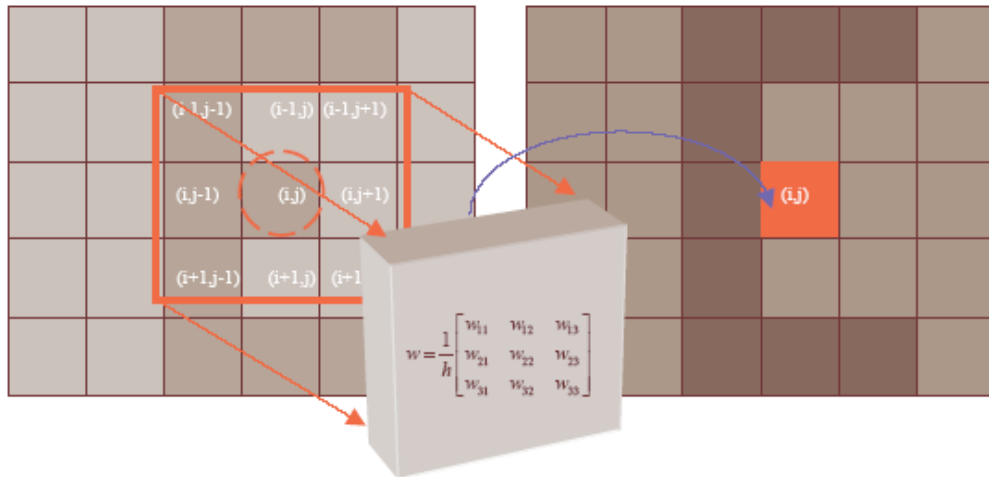


Figura 5.5: Filtrado espacial por una máscara de 3 x 3.

como la imagen resultante; si además consideramos a $w(r,t)$ la matriz que se utiliza para realizar el filtrado:

$$w = \frac{1}{h} \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}$$

donde $r=3$ y $t=3$, tendremos que cada elemento de $IT(m ; n)$ es calculado:

$$vt_{ij} = \frac{1}{h} [w_{22}vs_{ij} + w_{11}vs_{(i-1)(j-1)} + w_{12}vs_{(i-1)j} + w_{13}vs_{(i-1)(j+1)} \cdots \\ + w_{21}vs_{i(j-1)} + w_{23}vs_{i(j+1)} + w_{31}vs_{(i+1)(j-1)} + w_{32}vs_{(i+1)j} + w_{33}vs_{(i+1)(j+1)}]$$

La Figura 5.5 muestra estos detalles.



Figura 5.6: Imagen resultado del filtraje espacial.

1.7.1.6 Extracción de Bordos

En visión computacional es de utilidad para hacer reconocimiento de objetos o bien para segmentar regiones, extraer los bordes de objetos (que en teoría delimitan sus tamaños y regiones). La función *edge* da la posibilidad de obtener los bordes de la imagen. La función permite encontrar los bordes a partir de dos diferentes algoritmos que pueden ser elegidos, *canny* y *sobel*.



Figura 5.7: Imagen resultado de la aplicación del algoritmo de canny.

1.7.1.7 Imágenes Binarias y Segmentación por Umbral

Una imagen binaria es una imagen en la cual cada píxel puede tener solo uno de dos valores posibles 1 o 0. Como es lógico suponer una imagen en esas condiciones es mucho más fácil encontrar y distinguir características estructurales.

El trabajo con imágenes binarias es muy importante ya sea para realizar segmentación por intensidad de la imagen, para generar algoritmos de reconstrucción o reconocer estructuras.

La forma más común de generar imágenes binarias es mediante la utilización del valor umbral de una imagen a escala de grises; es decir se elige un valor límite (o bien un intervalo) a partir del cual todos los valores de intensidades mayores serán codificados como 1 mientras que los que estén por debajo serán codificados a 0.

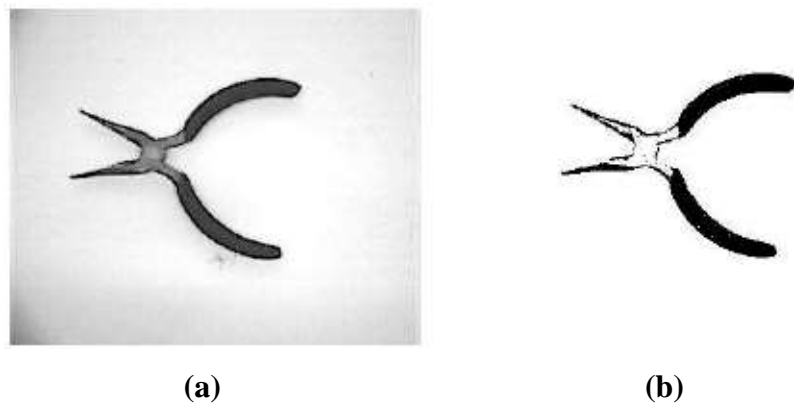


Figura 5.8: a) imagen original y b) imagen resultado de la aplicación de un umbral

1.7.1.8 Operaciones morfológicas

Una de las operaciones más utilizadas, sobre imágenes previamente binarizadas es las operaciones morfológicas. Las operaciones morfológicas son operaciones realizadas sobre imágenes binarias basadas en formas. Estas operaciones toman como entrada una imagen binaria regresando como resultado una imagen también binaria. El valor de cada píxel de la imagen binaria resultado es basado en el valor del correspondiente píxel de la imagen original binaria y de sus vecinos. Entonces eligiendo apropiadamente la forma de los vecinos a considerar, puede construirse operaciones morfológicas sensibles a una forma en particular.

Las principales operaciones morfológicas son la **dilatación** y la **erosión**. La operación de dilatación adiciona píxeles en las fronteras de los objetos, mientras la erosión los remueve. En ambas operaciones se utiliza una rejilla que determina cuales vecinos del elemento central de la rejilla serán tomados en cuenta para la determinación del píxel resultado. La rejilla es un arreglo cuadrangular que contiene *unos* y *ceros*, en los lugares que contiene *unos* serán los vecinos de la imagen original con respecto al píxel central, los cuales serán tomados en consideración para determinar el píxel de la imagen resultado, mientras que los lugares que tengan *ceros* no serán tomados en cuenta. La Figura 5.9 indica gráficamente el efecto de la rejilla sobre la imagen original y su resultado en la imagen final.

Como muestra la Figura 5.9 solo los píxeles de color amarillo en la imagen original participan en la determinación del píxel rojo de la imagen resultado.

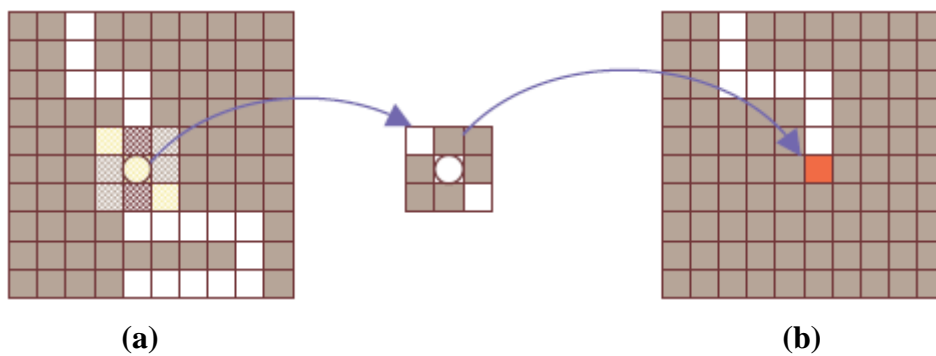


Figura 5.9: a) imagen original y b) imagen resultado de la aplicación de la operación morfológica considerando la rejilla de 3 x 3.

Una vez determinado el tamaño de la rejilla y su configuración. Se aplica la operación morfológica. En el caso de la dilatación, si alguno de los píxeles de la rejilla configurados como *unos* coincidió con *al menos uno* de la imagen original el píxel resultado es uno. Por el contrario en la erosión todos los píxel de la rejilla, configurados como *unos* deben coincidir con *todos* los de la imagen si esto no sucede el píxel es 0.

1.7.1.9 Operaciones Basadas en Objetos

En una imagen binaria, puede definirse un objeto como un conjunto de píxeles conectados con valor 1. Por ejemplo la Figura 5.10 representa una imagen binaria conteniendo un objeto cuadrado de 4 x 4. El resto de la imagen puede ser considerado como el fondo.

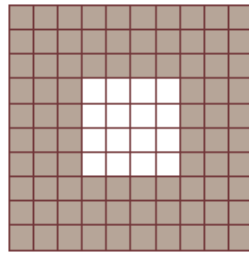


Figura 5.10: Imagen binaria conteniendo un objeto.

Una imagen indexada es una imagen que puede representar imágenes a color al igual que las RGB pero utiliza un formato diferente, en lugar de utilizar tres planos como lo realiza las imágenes RGB utiliza una matriz y una tabla, la matriz contiene en cada píxel un número entero correspondiente al índice de la tabla, mientras que cada índice de la tabla corresponden 3 valores correspondiente a los planos RGB, con ello es posible reducir el tamaño de las imágenes al reducir el número de diferentes colores.



Figura 5.11: Ejemplo de imagen indexada.

1.7.1.10 *Medición de Características*

En visión computacional es de particular interés encontrar mediciones de características tales como el área, centroide y otras de objetos previamente etiquetados o clasificados por la función `bwlabel` con el objetivo de identificar su posición en la imagen.

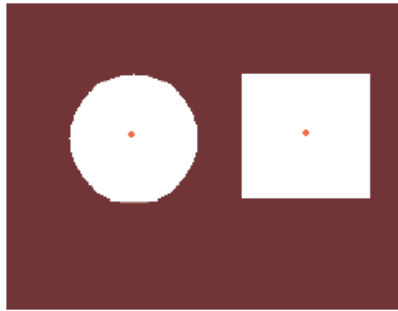


Figura 5.12: Identificación de los centroides.

1.7.1.11 *Imágenes y Formatos de Color*

El formato de representación de color ofrecido por las imágenes RGB resulta no apropiado para aplicaciones en las cuales el cambio de iluminación es problema. Otro tipo de formatos de color menos sensibles al cambio de iluminación han sido propuestos, tales como el modelo HSV. MATLAB dispone de funciones especiales para realizar cambios entre modelos de color y para convertir imágenes de color a escala de grises.



Figura 5.13: Imagen RGB.

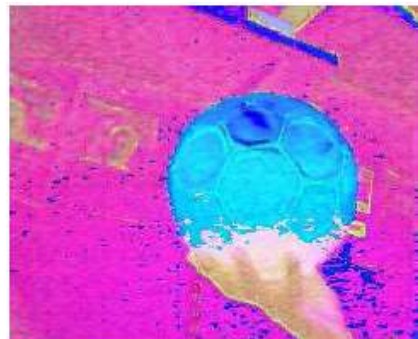


Figura 5.14: Imagen HSV.

1.7.2 Transformación de Fourier

Según el libro Image Processing ToolBox versión 6 User's Guide podemos recopilar esta información sobre la transformada de Fourier como cultura general, ya que con esta transformada incorporada en MATLAB, se puede realizar diferentes operaciones y procesos con las imágenes, e indispensable saber a que se refiere este teorema. El teorema de Fourier afirma que una gráfica o función, cualquiera que sea su forma, se puede representar con alta precisión dentro de un intervalo dado, mediante la suma de una gran cantidad de funciones senoidales, con diferentes frecuencias. Dicho de otro modo, cualquier función, sea o no sea periódica, se puede representar por una superposición de funciones periódicas con diferentes frecuencias.

La variación de la irradiancia o brillantez de una imagen, medida a lo largo de una dirección cualquiera es entonces una función que se puede representar mediante el teorema de Fourier, con una suma de distribuciones senoidales de varias frecuencias. Sin entrar en detalles técnicos innecesarios, simplemente afirmaremos aquí que atenuar o reforzar individualmente algunas de estas componentes senoidales puede tener un efecto dramático en la calidad de una imagen, mejorándola o empeorándola, según el caso. Este es el fundamento del procesamiento de imágenes.

La transformada de Fourier, $F(u)$, de una sola variable, de una función continua, $f(x)$, se define como

$$F(u) = \int_{-\infty}^{+\infty} f(x) \cdot e^{-j2\pi \cdot u \cdot x} dx$$

De la misma forma, dada $F(u)$, podemos obtener $f(x)$ a través de la transformada inversa de Fourier

$$f(x) = \int_{-\infty}^{+\infty} F(u) \cdot e^{j2\pi \cdot u \cdot x} du$$

Las expresiones anteriores pueden extenderse fácilmente a dos variables, u y v :

$$F(u, v) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) \cdot e^{-j2\pi \cdot (u \cdot x + v \cdot y)} dx dy$$

Y, similarmente, la transformada inversa queda definida como

$$f(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F(u, v) \cdot e^{j2\pi \cdot (u \cdot x + v \cdot y)} du dv$$

Ahora bien, en el procesamiento de imágenes se trata con funciones discretas. Por esta razón, presentamos la transformada de Fourier de una función discreta (TDF) de una variable, $f(x)$, con $x=0, 1, \dots, M-1$:

$$F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) \cdot e^{-\frac{j2\pi \cdot u \cdot x}{M}}$$

Para $u=0, 1, \dots, M-1$. Análogamente, dada $F(u)$, la transformada discreta de Fourier inversa está definida como

$$f(x) = \sum_{u=0}^{M-1} F(u) \cdot e^{\frac{j2\pi \cdot u \cdot x}{M}}$$

Para $x=0, 1, \dots, M-1$. Cada uno de los M términos de $F(u)$ es llamado componente de frecuencia de la transformada. Los valores de u en los que la $F(u)$ está definida se llama dominio de la frecuencia.

Una propiedad importante de la TDF (y su inversa) es que, a diferencia del par continuo, aquéllas siempre existen. Lo mismo sucede con la TDF para una función de dos variables. En algunos casos, es conveniente expresar la TDF, $F(u)$, en coordenadas polares:

$$F(u) = |F(u)| \cdot e^{-j\phi(u)}$$

donde

$$|F(u)| = \sqrt{R^2(u) + I^2(u)}$$

es llamado *magnitud* de la transformada de Fourier, y

$$\phi(u) = \tan^{-1} \left[\frac{I(u)}{R(u)} \right]$$

se denomina *fase* o *argumento* de la transformada de Fourier. $R(u)$ e $I(u)$ son la parte real y la imaginaria de $F(u)$, respectivamente.

1.7.3 Histogramas

En la página de internet www.wikipedia.com las siguientes definiciones sobre histogramas, extracción de rasgos, textura y segmentación están definidas de una manera clara. Empezando por definir el histograma como: que el histograma de una imagen representa la frecuencia relativa de los niveles de gris de la imagen. Las técnicas de modificación del histograma de una imagen son útiles para aumentar el contraste de imágenes con histogramas muy concentrados.

Un histograma de intensidad muestra cómo los niveles de brillo individuales están ocupados en una imagen; el contraste de la imagen es moderado por el rango de niveles de brillo. El histograma traza el número de píxeles con un nivel de brillo particular contra el nivel de brillo. Para los píxeles de 8-bit, el brillo va de cero (negro) a 255 (blanco). Un histograma, además, puede mostrar que nosotros no hayamos usado todos los niveles grises disponibles. Sin embargo, podemos estirar la imagen para usarlos todos, y la imagen se pondría más clara. Sin duda, ésta es la atención esencialmente cosmética para poner bien la apariencia de la imagen.

Haciendo bien la apariencia, sobre todo en la vista del posterior procesamiento, es el enfoque de muchos funcionamientos de procesamiento de imagen básicos. Un histograma también puede revelar si hay ruido en la imagen, si el histograma ideal es conocido. Es necesario quitar este ruido, no sólo para mejorar la apariencia de la imagen, también, para aliviar más tarde la tarea de las técnicas de extracción de rasgos.

1.8 Extracción de Rasgos

El nivel alto de extracción de rasgos involucra, el hallazgo en la forma de imágenes de la computadora. Para poder reconocer las caras automáticamente, por ejemplo: un acercamiento, es extraer los rasgos del componente. Esto requiere la extracción de: por decir, los ojos, los oídos, y la nariz, qué son los rasgos mayores

de la cara. Para encontrarlos, podemos usar su forma: la parte blanca de los ojos es elipsoidal; la boca puede aparecer como dos líneas, como también las cejas. La extracción de la forma implica el hallazgo de su posición, su orientación y su tamaño. Este proceso de extracción de rasgos puede verse, similar a la manera que nosotros percibimos el mundo: muchos libros para los bebés describen las formas geométricas básicas como los triángulos, círculos y cuadrados. Los cuadros más complejos pueden descomponerse en una estructura de formas simples. En muchas aplicaciones, a propósito, el análisis puede guiarse en las formas como están colocadas. Para el ejemplo de análisis de imagen de la cara, nosotros esperamos encontrar los ojos anteriormente, y lateral de la nariz y esperamos encontrar la boca debajo de la nariz.

1.9 Introducción a Textura y Segmentación

1.9.1 Textura

Cabe mencionar que muchos diccionarios definen a textura como; la propiedad que tienen las superficies externas de los objetos, así como las sensaciones que causan, que son captadas por el sentido del tacto. Pero textura realmente posee un concepto muy nebuloso, a menudo es atribuido como a la percepción humana, o la apariencia de tejido. Todos tenemos nuestra propia interpretación acerca de la naturaleza de textura; no hay ninguna definición matemática para la textura, simplemente existe:

Textura.- Es el arreglo de hilos, etc., en tejido textil. Entonces la percepción característica del elector debido a esto sería: el arreglo de pequeñas partes, que percibe una estructura, (de piel, piedra, tierra, tejido orgánico, trabajo literario, etc.); la representación de la estructura y el detalle de objetos en el arte.

Esta definición comprende mucho. Si nosotros cambiamos los 'hilos' por 'píxeles' entonces la definición podría ser aplicada a las imágenes (salvo la parte sobre obra de arte).

Como una definición alternativa de textura, se puede considerar como una base de datos de imágenes que las investigaciones usan para probar algoritmos.

1.9.2 Segmentación

Según el libro Image Processing ToolBox versión 6 User's Guide, cuando ya se dispone de la imagen capturada y filtrada, es necesario aislar o separar los objetos de interés de la escena. Se pretende por tanto dividir una imagen en diferentes regiones, o dicho de otra forma, detectar automáticamente los bordes entre los elementos o regiones. Las operaciones de segmentación de una escena dependen de la propia escena y de la información que se busque dentro de la imagen. Por ejemplo, en fotografía aérea se utiliza para separar regiones pobladas, agrícolas, bosques, aguas, etc. En imagen médica se usa para aislar zonas enfermas de los órganos, o para detectar y contar microorganismos en microscopia. Industrialmente, se usa como primer paso para la medición de piezas, para determinar la correcta disposición del etiquetado de los envases (aislar el envase, aislar la etiqueta, determinar si la relación entre estos dos objetos es correcta), para la medición de irregularidades en superficies homogéneas, para la detección de cambios de texturas en las que el sistema ha de determinar si es un defecto o no lo es, etc.

Las técnicas básicas de segmentación se pueden dividir en tres grupos:

- Aplicación de umbrales de niveles de gris.
- Agrupación por rasgos comunes.
- Extracción de bordes.

NOTA: Usando el umbral óptimo, se realiza la segmentación de la imagen. Si el objeto y el fondo difieren en el nivel de gris, se pueden identificar los píxeles que pertenecen al objeto mediante el proceso de segmentado. Para segmentar la imagen se utiliza la binarización: se etiqueta con cero los píxeles cuyo nivel de gris sean menores al valor del umbral óptimo y con 1 a los píxeles con nivel de gris mayor.

“Quien no se resuelve a cultivar el hábito de pensar, se pierde el mayor placer de la vida”

Thomas A. Edison.

CAPITULO II

“DESCRIPCION, ANALISIS E INTERPRETACION DE RESULTADOS”

2.1 Breve descripción de la Universidad Técnica de Cotopaxi

En la Guía Académica de la Universidad Técnica de Cotopaxi desarrollada por el Ing. M.Sc. Hernán Yánez y otorgada a los estudiantes en la matrícula, obtenemos una pequeña reseña histórica del origen de nuestra Universidad, sus ofertas y lo que ha logrado hasta ahora a partir de su creación, resaltando lo siguiente:

La Universidad Técnica de Cotopaxi, es una Institución de Educación Superior Pública, Laica, y gratuita, creada mediante ley promulgada en el Registro Oficial N.- 618 del 24 de Enero de 1995, y que forma parte del Sistema Nacional de Educación Ecuatoriano. Se rige por la Constitución Política del Estado, la ley de Educación Superior y otras leyes conexas. Es una institución universitaria sin fines de lucro que orienta su trabajo hacia los sectores urbanos, marginales y campesinos; que busca la verdad y la afirmación de la identidad nacional, y que asume con responsabilidad el aseguramiento de la libertad en la producción y difusión de los conocimientos y del pensamiento democrático y progresista para el desarrollo de la conciencia antiimperialista del pueblo.

En la institución se forman actualmente profesionales en las siguientes áreas de especialidades: Ciencias Aplicadas; Ciencias Agropecuarias, Ambientales y Veterinarias; Ciencias Humanísticas y del Hombre. Por ello, la Universidad Técnica de Cotopaxi asume su identidad con gran ímpetu y honorabilidad: “Por la vinculación de la universidad con el pueblo”, “Por una Universidad alternativa con visión de Futuro”.

2.1.1 Descripción de la Especialización de Diseño Gráfico

Según la pagina web www.utc.edu.ec/ el comienzo del diseño gráfico se remonta al invento y difusión de la imprenta, mientras que el diseño de moda y el industrial se desarrollaron a la par que las revoluciones industriales.

En la Universidad Técnica de Cotopaxi y en otras universidades, la Especialización en Diseño Gráfico Computarizado, permite en el profesional que curse dicha rama; el desarrollo de destrezas, habilidades con pensamiento crítico e innovador, ser competitivo, integral, defensor y difusor de la ciencia y la tecnología; además de estar capacitado para resolver problemas y necesidades de la información y comunicación visual. Planifica, crea y produce las tareas de diseño; utiliza los medios gráficos e imágenes para la anunciación de la información, en otras palabras, preparado para transmitir mensajes efectivos con principios visuales computarizados.

Un ingeniero en Diseño Gráfico computarizado está adecuado para prestar su actividad profesional, tanto en el sector público y privado, o independientemente como diseñador de productos gráficos en las áreas de Industrias Gráficas y Editoriales, Publicidad, Sitios Web, Dibujante Ilustrativo, Director de Arte Gráfico, etc.

2.1.2 Análisis de la Importancia de las Imágenes en Diseño Gráfico

Las habilidades, destrezas y capacidades cultivadas en los estudiantes de la Especialización de Diseño Gráfico Computarizado son resultado de las múltiples materias recibidas en cada semestre en el transcurso de la carrera; en las que se puede estudiar, manipular y tratar imágenes. Las mismas que son, entre otras: Teoría de la imagen y Cromática, Fotografía básica, Dibujo artístico, Taller del diseño, Fotografía publicitaria, Computación y Graficación, etc. Con lo que cabe decir que la especialización de diseño gráfico computarizado tiene como base central, una imagen o un gráfico y una computadora.

2.1.2.1 Formatos de gráficos más importantes

El grupo investigador afirma que en la especialización de diseño gráfico se sabe de la importancia que tiene una imagen digital en su entorno; dicha razón ha generado en el mundo, la ciencia y la tecnología la creación de múltiples formatos de imágenes y diversos tipos de imágenes, como también sobre el color; por esa misma razón es indispensable tener un conocimiento de esos temas, y en los siguientes párrafos nos referiremos a ellos poniendo en claro conceptos básicos:

Según la página web www.wikipedia.org :

Formato de Imagen BMP (Bitmapped File Format): Junto con el surgimiento de Windows 3.x se desarrolla un nuevo formato gráfico bitmap que constituye el estándar adoptado por este entorno operativo, es decir al BMP, en el cual están almacenadas las imágenes que constituyen los llamados *wallpapers*. Este formato guarda las imágenes descomprimidas, lo que significa mayor velocidad de carga y mayor espacio requerido.

Formato de Imagen GIF (Graphic Interchange Format): Es el formato gráfico bitmap por excelencia. Fue creado por COMPUSERVE en junio de 1987 y con el paso del tiempo se ha convertido en el formato más difundido en el mundo.

A la primera versión se la llamó GIF 87a, y a la segunda, GIF 89a. Esta última versión presenta nuevas características para facilitar el manejo de imágenes en este formato. Los GIFs utilizan una paleta de entre 2 y 256 colores. Poseen una rutina de compresión muy eficaz que, aunque demora un poco la carga, reduce los archivos a una tamaño mucho menor que otros formatos. Gracias a esa rutina de compresión que empequeñece los archivos, el GIF es el formato óptimo para ser bajado de BBS o Internet. La resolución máxima alcanzada es la de 1024 x 768 píxeles en 256 colores, pero no hay razón por la cual no pueda crearse una imagen de mayor tamaño.

Formato de Imagen JPG (JPEG): El formato JPEG ofrece los imprescindibles 16 millones de colores (truecolor), unido a una compresión realmente asombrosa. Sólo tiene una limitación: para obtener esos valores de compresión modifica sutilmente la imagen, descartándose su uso en aplicaciones en las que se desea mantener una calidad bit a bit. El diseño de este formato está pensado para almacenar imágenes del "mundo real", también llamadas imágenes de tono continuo, como digitalizaciones o renderizaciones de alta calidad. Si se intenta almacenar imágenes de tipo vectorial o dibujos sencillos no realísticos, se observará como la compresión disminuye enormemente, y las modificaciones hechas sobre la imagen original por el algoritmo de compresión se observan a simple vista. La abreviación JPEG viene de las iniciales de **Joint Photographic Experts Group**. Se trata del grupo de expertos que definieron las bases de este formato. JPEG, junto con GIF, son los formatos de imágenes usados en WWW.

Formato de Imagen TIFF (Tagged Image File Format - TIFF) O TIF: Más que una imagen es un archivo, el formato TIF (formato de archivo de imagen etiquetada) contiene una serie de bloques que conforman la imagen. Estos bloques pueden contener cierta información sobre la imagen en sí, su tamaño, su manejo del color, información a las aplicaciones que utilicen ese archivo, texto, y hasta thumbnails.

Un thumbnail o miniatura es una pequeña representación de una imagen mucho más extensa, a la cual el programa accede rápidamente y no pierde tiempo

descomprimiendo toda la imagen. Sirve para ver el contenido del archivo de una manera rápida y segura.

Fue introducido por Aldus Corporation y es el formato más usado cuando se trabaja con scanner debido a su útil manejo del color.

2.1.2.2 Fundamentos del color

Los colores primarios son el rojo (R), verde (G), y azul (B). Pueden ser sumados para producir los *colores secundarios* de la luz: magenta (rojo mas azul), cian (azul mas verde) y amarillo (rojo mas verde). Mezclando los 3 primarios, o un secundario con su primario opuesto en las intensidades correctas, podemos obtener luz blanca.

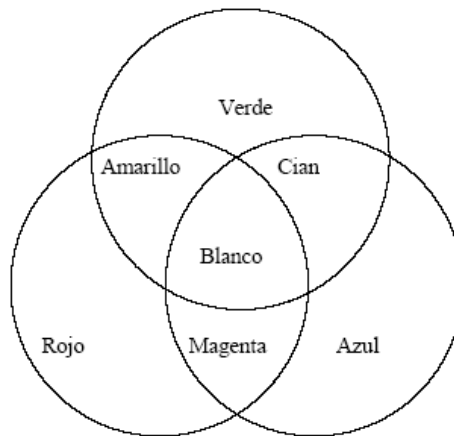


Figura 6.0: Suma de los colores primarios.

2.1.2.3 Modelos de color

El propósito de un modelo de color (también llamado espacio de color o sistema de color) es facilitar la especificación de colores de un modo estándar o aceptado en general. Un modelo de color es la especificación de un sistema de coordenadas y el subespacio dentro de él donde cada color puede ser representado por un solo punto. La mayoría de los modelos de color en uso se orientan ya sea al hardware, o a aplicaciones donde la manipulación de color es el objetivo. Los modelos orientados a hardware más comunes son el RGB (rojo, verde, azul), para

monitores a color y cámaras de video; el CMY (cian, magenta, amarillo) y CMYK para impresión a color; y el HSI (tono, saturación, intensidad), que se acerca a la manera como el ojo humano describe e interpreta el color.

Modelo RGB.- Se basa en la combinación de tres señales de luminancia cromática distinta: rojo, verde, azul (*Red, Green, Blue*). La forma más sencilla de obtener un color específico es determinar la cantidad de color rojo, verde y azul que se requiere combinar para obtener el color deseado; para lo cual se realiza la suma aritmética de las componentes: $X = R + G + B$, gráficamente representada por un cubo.

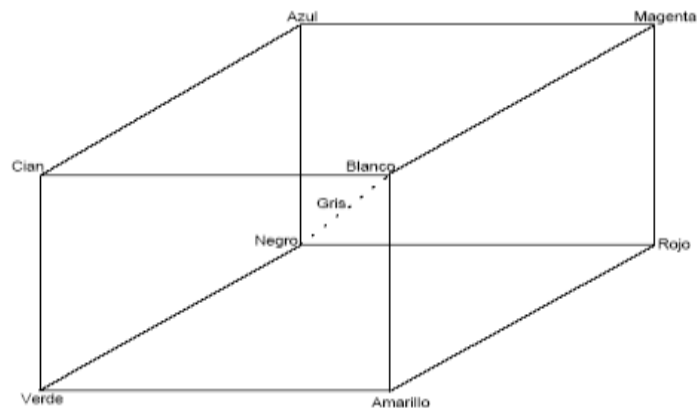


Figura 6.1 Espacio del modelo de color RGB.

2.1.2.4 Mapa de color

El mapa de color es una matriz de $n \times 3$, donde cada renglón es una tripleta de colores. El primer renglón corresponde al valor mínimo del eje de color y el último renglón al máximo. Al definir diferentes distribuciones de intensidad de los tres colores básicos se crean diferentes mapas de color. Algunos de los mapas de color predeterminados en MATLAB son:

hsv, cool, hot, jet, gray, flag

2.2 Aplicaciones sobre el Procesamiento de Imágenes

Según el libro de Winston P. "Inteligencia Artificial" define a los siguientes términos de una manera clara, lo que motiva al entendimiento de aspectos necesarios en la consecución del software prototipo, mención que refiere a la visión por computadora, que es una técnica basada en la adquisición de imágenes, generalmente en dos dimensiones, para luego procesarlas digitalmente mediante algún tipo de CPU (computadora, microcontrolador, DSP, etc.), con el fin de extraer y medir determinadas propiedades de las imágenes adquiridas. Se trata, por tanto, de una tecnología que combina las computadoras con las cámaras de video para adquirir, analizar e interpretar imágenes de una forma equivalente a la inspección visual humana.

Según el grupo investigador actualmente se aplica en diversos procesos científicos, de ingeniería y militares, extendiéndose su uso además, en un amplio rango de sectores industriales para la automatización de tareas anteriormente reservadas para la inspección visual humana. Las técnicas de visión por computadora, son particularmente apropiadas para la realización de trabajos visuales altamente repetitivos que sean fatigosos o difíciles de realizar para un operario.

La página de Internet [http://www.aitex.es/estudios/Vision Artificial1.pdf](http://www.aitex.es/estudios/Vision%20Artificial1.pdf), comenta que hoy en día casi no hay áreas de enfoque técnico que no hayan sido impactadas de alguna manera por el procesamiento digital de imágenes. Las áreas son tan abundantes y variadas que se tratará un criterio de clasificación para dar algunos ejemplos. En este caso el criterio de clasificación será por la fuente de energía de las imágenes; mostrada por el espectro electromagnético (otras fuentes son la acústica, la ultrasónica, y la electrónica). Una de las mejores clasificaciones de aplicaciones que se le puede acotar a la Visión Artificial y por tal motivo al procesamiento de imágenes y sus procesos a fines, es esta mencionada por este sitio de Internet.

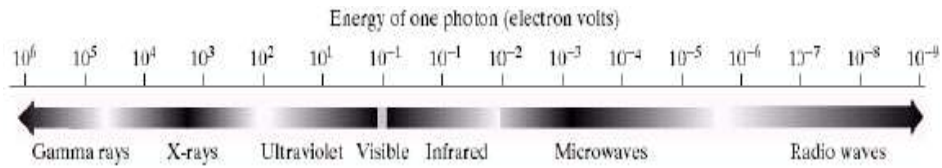


Figura 7.0: Espectro EM en energía por fotón.

2.2.1 Rayos Gamma

Sus aplicaciones más importantes: la medicina nuclear y las observaciones astronómicas.

Medicina nuclear: Se inyecta al paciente un isótopo radioactivo que emite rayos gamma cada vez más débiles, estos se detectan y se forma la imagen. Otro caso es la Tomografía de Emisión de Positrones (PET), en esta se inyecta al paciente un isótopo radioactivo que emite positrones. Los positrones chocan con los electrones del paciente, se destruyen ambos, y se producen 2 rayos gamma que se detectan para formar la imagen.



Figura 8.0: Tomografía por emisión de positrones (PET)

Astronomía: Se detecta la radiación natural de rayos gamma de los astros.



Figura 8.1: Imagen a partir de rayos gamma de Cygnus Loop

2.2.2 Rayos X

Los rayos X no sólo se usan en medicina, sino también en gran manera en la industria y otras áreas, como la astronomía.

Angiografía: Es una aplicación de la radiografía de mejora de contraste. Se utiliza para obtener angiogramas, imágenes de venas y arterias. Un cateter se inserta en una de estas y se guía hasta el área a estudiar. Una vez ahí, se inyecta por medio de éste un medio que mejora el contraste y permite al radiólogo ver irregularidades o bloqueos.



Figura 9.0: Angiograma de la aorta

Tomografía axial computarizada (CAT)

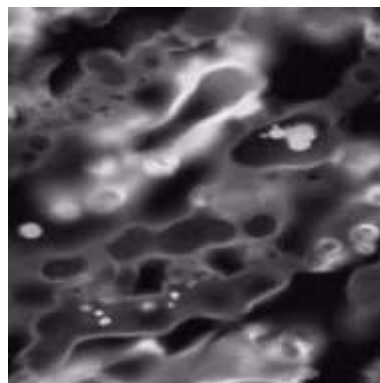


Figura 9.1: Tomografía de la cabeza

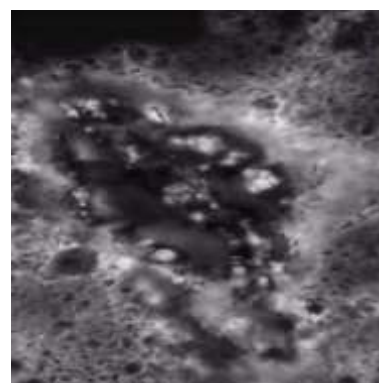
2.2.3 Ultravioleta

Las aplicaciones incluyen litografía, inspección industrial, microscopía, lasers, imágenes biológicas y observaciones astronómicas.

Microscopía de fluorescencia: Al dirigir luz ultravioleta a un mineral con fluor se observa que este brilla. La microscopía de fluorescencia utiliza este principio, dirigiendo una luz de excitación a un objeto preparado y separando la radiación ultravioleta producida (débil) de la producida por la luz de excitación (brillante).



a) Maíz sano



b) Maíz enfermo

Figura 10.0: Ejemplos de imágenes ultravioleta

2.2.4 Visible e Infrarroja

Estas son, por mucho, las aplicaciones más numerosas. La banda infrarroja se utiliza usualmente en conjunto con la imagen visual.

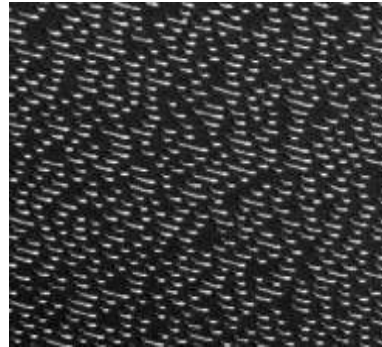


Figura 11.0: Superficie de un CD de audio al microscopio

Detección remota: Usualmente incluye varias bandas que detectan diferentes detalles.

Banda	Nombre	Longitud de onda	Usos y características
1	Azul visible	0.45 – 0.52	Penetración máxima del agua
2	Verde visible	0.52 – 0.60	Bueno para medir la salud de las plantas
3	Rojo visible	0.63 – 0.69	Discriminación de vegetación
4	Infrarrojo bajo	0.76 – 0.90	Mapeo de costas y biomasa
5	Infrarrojo medio	1.55 – 1.75	Contenido de humedad en el suelo y vegetación
6	Infrarrojo termal	10.4 – 12.5	Humedad del suelo y mapeo termal
7	Infrarrojo alto	2.08 – 2.35	Mapeo mineral



(1)

(2)

(3)

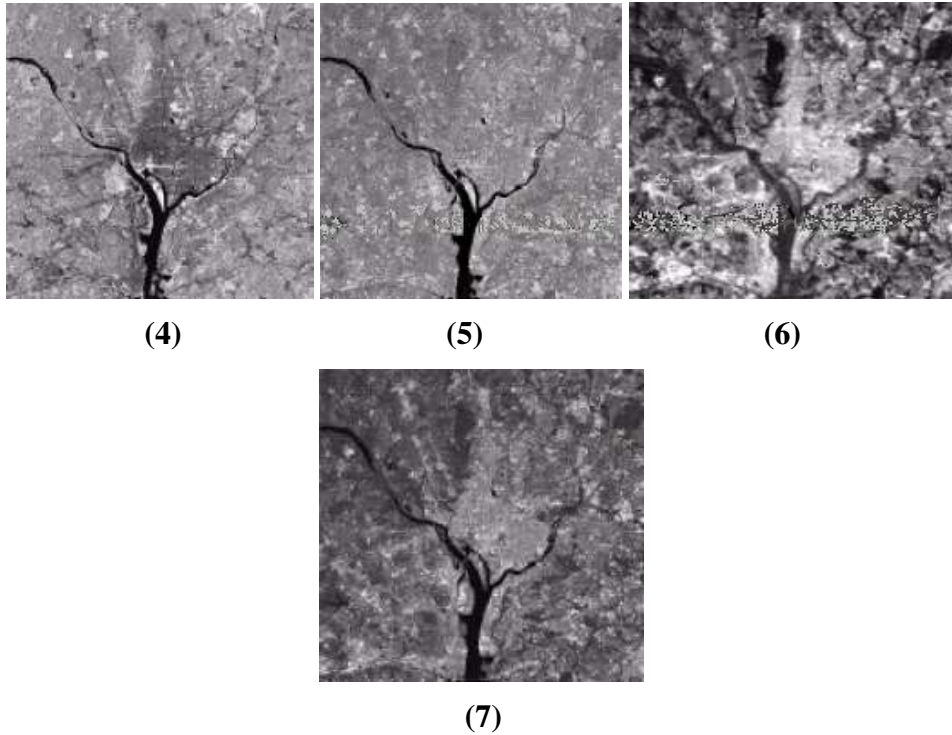


Figura 11.1: Bandas del satélite LANDSAT

NOTA: Estas imágenes pueden ser usadas para ver, en ciudades, movimientos o crecimiento de población, contaminación, u otros factores que dañan al ambiente. Otras aplicaciones incluyen observación y predicción del clima.

Control de calidad: Una aplicación muy utilizada de la luz visible e infrarroja es en el control de calidad, donde se utiliza para revisar productos de orden variado (niveles correctos en botellas, circuitos integrados con todos los componentes, errores de fabricación o material defectuoso).

En este mundo tan automatizado,, el control de calidad a veces se somete a estrategias bastante desactualizadas, que provocan al final pérdidas en las empresas antes que la mejora de la misma. El término Visión Artificial en el control de calidad es lo más común que se puede notar en empresas de primer orden que invierte de forma segura este tipo de sistemas.

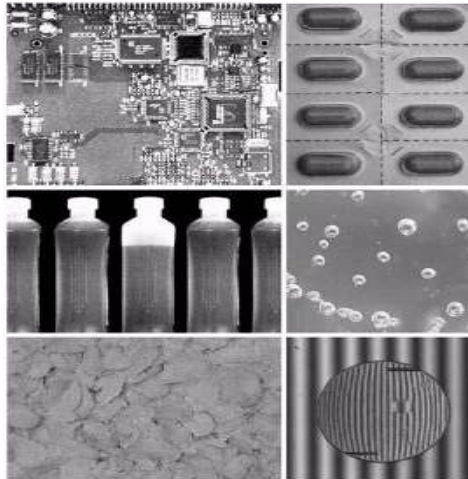


Figura 11.2: Control de calidad en circuito integrado, empaque de píldoras, nivel de líquidos, burbujas en plástico transparente, cereal, y fallas en una lente intraocular.

2.2.5 Microondas

Su aplicación principal es el radar. El radar puede obtener datos, virtualmente de cualquier región a cualquier hora, sin importar condiciones de clima o de luz ambiental (puede penetrar nubes, vegetación, hielo, arena). Un radar utiliza su propia “iluminación” (pulsos de microondas) para obtener una imagen resultante de la energía reflejada.

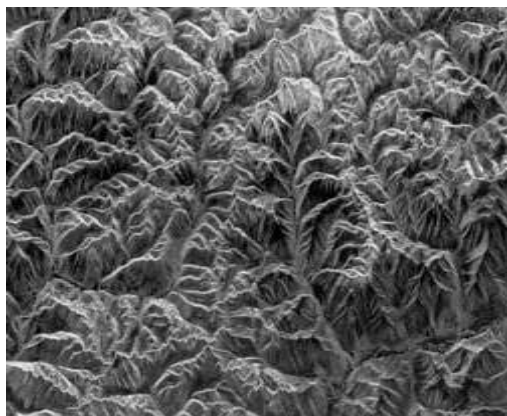


Figura 12.0: Imagen de radar del tibet.

2.2.6 Ondas de Radio

Mayoritariamente se usa en medicina y astronomía.

Medicina: Resonancia magnética (MRI). Se introduce al paciente en un imán de gran poder y se pasan ondas de radio por su cuerpo en pulsos cortos. Cada pulso causa un pulso resultante de ondas de radio desde los tejidos del paciente. La computadora determina el lugar y fuerza de estos pulsos, y produce una imagen bidimensional de una sección del paciente.

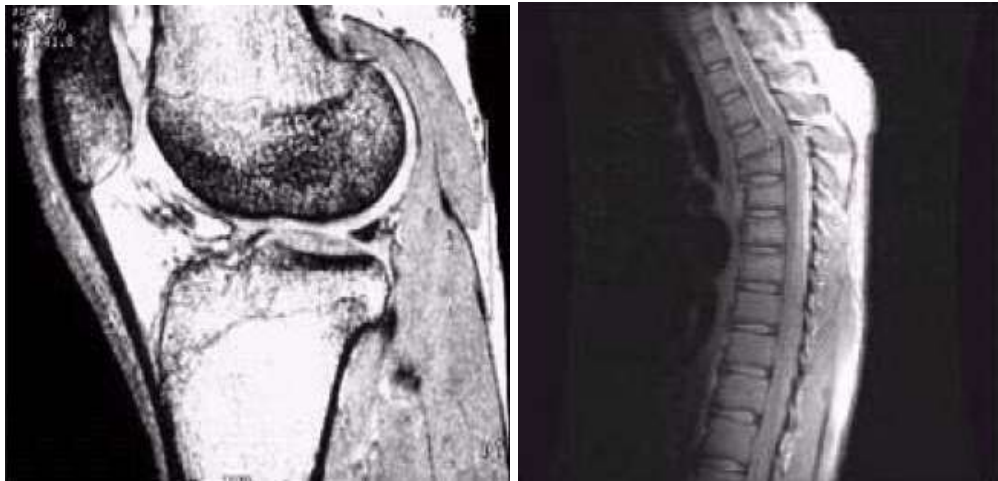


Figura 13.0: Imágenes MRI de una rodilla y una espina.

2.2.7 Aplicaciones de otras Fuentes de Energía

Sonido: Se aplica en exploración geológica (petróleo y minerales), la industria, y medicina. En la primera se utiliza sonido en la parte baja del espectro (cientos de Hertz), y las otras áreas utilizan ultrasonido. Por ejemplo, en geología se utiliza un gran camión para presionar una placa en el suelo, y se hace vibrar el camión a una frecuencia de cerca de 100 Hz. Las ondas que regresan se analizan. Un método similar se utiliza para adquisición submarina, al disparar 2 pistolas de aire y recuperar las ondas de regreso para producir un mapa 3D del fondo del océano.

2.3 Análisis de resultados de la encuesta desarrollada por los Estudiantes y Docentes de Diseño Gráfico

De la indagación realizada a las estudiantes y docentes de la especialización en Diseño Gráfico, a través de un solo tipo de encuesta con nueve preguntas aplicadas a una población de cien individuos; participes en la recopilación de datos indispensables para justificar el desarrollo del software prototipo, se obtuvieron los siguientes resultados:

1. Para usted una imagen digital es:

Cuadro N° 2.0: Definición de imagen digital

DEFINICIÓN DE IMAGEN DIGITAL		
OPCIÓN	f	%
a) Aquella que, se compone de un número finito de elementos, cada uno con un lugar y valor específicos. Estos elementos son llamados <i>pels</i> , o <i>píxeles</i> .	79	79
b) Una instrucción de como colorear cada píxel de un display o pantalla.	6	6
c) La reproducción de la figura de un objeto por la combinación de los rayos de luz que proceden de él, a través de una pantalla.	13	13
d) Ninguna de las definiciones anteriores.	1	1
e) No conoce alguna definición.	1	1
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

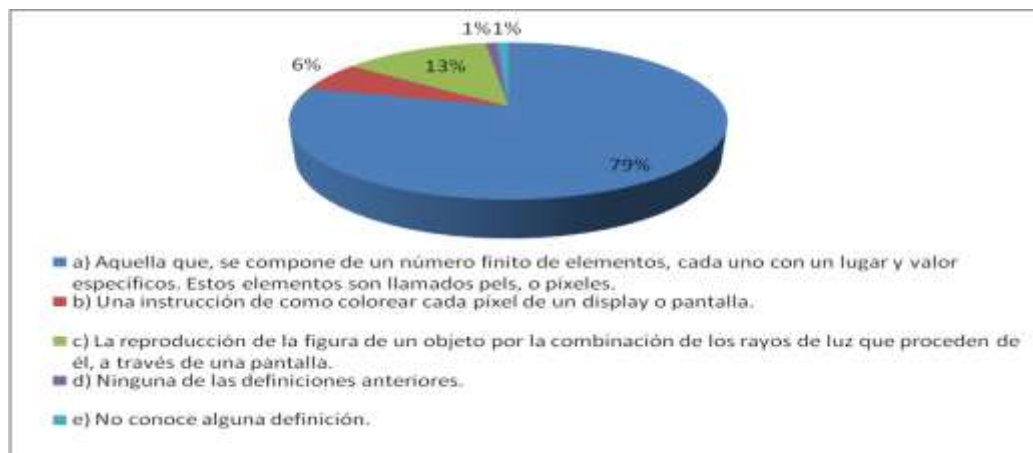


Gráfico N° 1: Definición de imagen digital.

Síntesis Pregunta 1: El poseer un concepto para imagen digital es de vital importancia en la apreciación del software prototipo para procesamiento de imágenes y sus utilidades. Satisfactoriamente para el grupo investigador, los resultados son los esperados al tener un 79% de los encuestados con una definición exacta de imagen digital y un 13%, con una definición complementaria.

2.- Piensa que el tratamiento de una imagen digital, dentro del diseño gráfico computarizado, tiene prioridad:

Cuadro N° 3.0: Importancia del PDI en Diseño Gráfico

IMPORTANCIA DEL PDI EN DISEÑO GRÁFICO		
OPCIÓN	f	%
a) Alta.	88	88
b) Media.	10	10
c) Baja.	0	0
d) No tiene importancia.	1	1
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

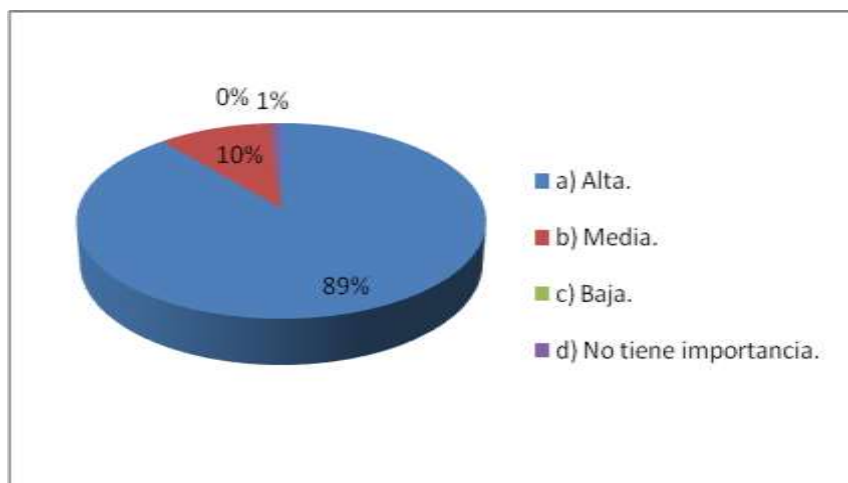


Gráfico N° 2: Importancia del PDI en Diseño Gráfico.

Síntesis Pregunta 2: Los datos obtenidos a partir de las encuestas; aplicadas a los estudiantes y docentes de la especialización en Diseño Gráfico, dan apertura y visto bueno a la implementación del software prototipo para procesamiento de imágenes y mejoramiento de las mismas; al mostrar con un 88% la alta importancia del procesamiento de imágenes digitales dentro de su especialización.

3.- ¿Qué entiende usted por procesar una imagen digital?

Cuadro N° 4.0: Entendimiento acerca del procesamiento de imágenes

ENTENDIMIENTO ACERCA DEL PROCESAMIENTO DE IMÁGENES		
OPCIÓN	f	%
a) Trabajar con una imagen digital, cambiar y mejorar su aspecto.	36	36
b) Modificar las características en los datos de una imagen digital.	9	9
c) Tener los datos de una imagen digital para poder trabajar con ellos, cambiando y mejorando su aspecto.	55	55
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador



Gráfico N° 3: Entendimiento acerca del procesamiento de imágenes.

Síntesis Pregunta 3: En la pregunta tres la información proporcionada por las encuestas, concluyen que el 55% de los encuestados, afirman que para ellos, procesar una imagen digital es; tener los datos de una imagen digital para poder trabajar con ellos, cambiando y mejorando su aspecto. Congeniadamente a un 36% de la opción (a) que presenta una similar concepción.

4.- Para usted un computador es:

Cuadro N° 5.0: Definición de Computadora

DEFINICIÓN DE COMPUTADORA		
OPCIÓN	f	%
a) Una máquina electrónica que facilita el trabajo del usuario.	76	76
b) Una máquina que trabaja con información, cálculos y tareas de texto.	24	24
c) Una máquina para el entretenimiento y multimedia.	0	0
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

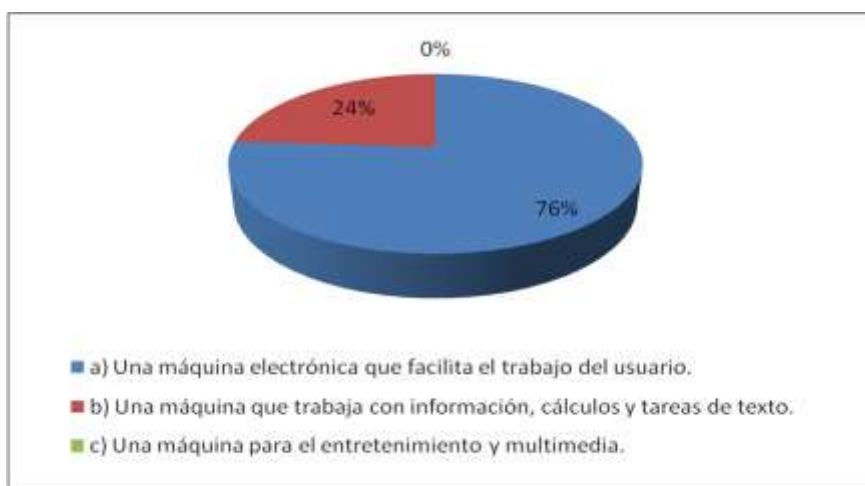


Gráfico N° 4: Definición de Computadora.

Síntesis Pregunta 4: La información obtenida para la pregunta cuatro, por medio de las encuestas planteadas, dan como resultado que el 76% de los encuestados de la carrera de Diseño Gráfico, saben que, un ordenador es una máquina electrónica que facilita el trabajo del usuario. También que procesar y obrar con una imagen, no puede ser la excepción.

5.- En su especialización, usted ha procesado imágenes digitales:

Cuadro N° 6.0: Procesamiento de Imágenes Digitales

PROCESAMIENTO DE IMÁGENES DIGITALES		
OPCIÓN	f	%
a) Si	95	95
b) No	5	5
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

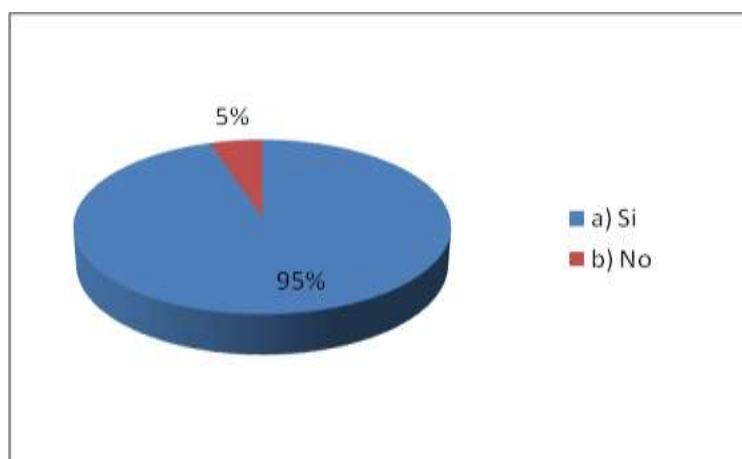


Gráfico N° 5: Procesamiento de Imágenes Digitales.

Síntesis Pregunta 5: En su mayoría todas las personas hemos procesado imágenes digitales, en especial aquellas personas que estamos vinculadas al uso de un computador como herramienta de trabajo o de estudio; aunque sin darnos cuenta o sin conocer el término técnico, lo hemos hecho. Los resultados de las encuestas son claros, al reportar al 95% de los encuestados como participes en esta actividad.

6.- Sus destrezas para maniobrar programas de computadora son:

Cuadro N° 7.0: Destreza para maniobrar programas

DESTREZA PARA MANIOBRAR PROGRAMAS		
OPCIÓN	f	%
a) Excelentes.	19	19
b) Buenas.	62	62
c) Regulares.	19	19
d) Le es muy difícil.	0	0
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

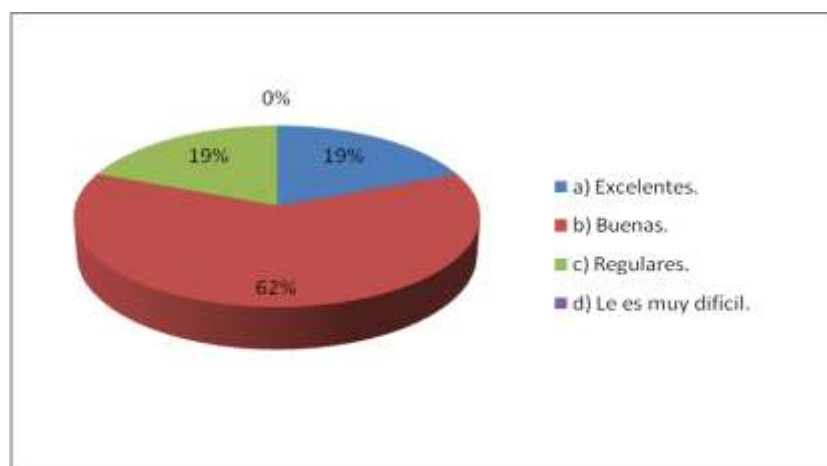


Gráfico N° 6: Destreza para maniobrar programas.

Síntesis Pregunta 6: Las estadísticas para la pregunta sexta, planteada en la encuesta, es ventajosamente favorable, ya que señala que el 62% de los encuestados, tienen buenas destrezas para trabajar y maniobrar con programas de computadora.

7. Utiliza un software para procesar imágenes digitales:

Cuadro N° 8.0: Uso de un software para procesar imágenes

USO DE UN SOFTWARE PARA PROCESAR IMÁGENES		
OPCIÓN	f	%
a) Siempre.	66	66
b) A veces.	33	33
c) Nunca.	1	1
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

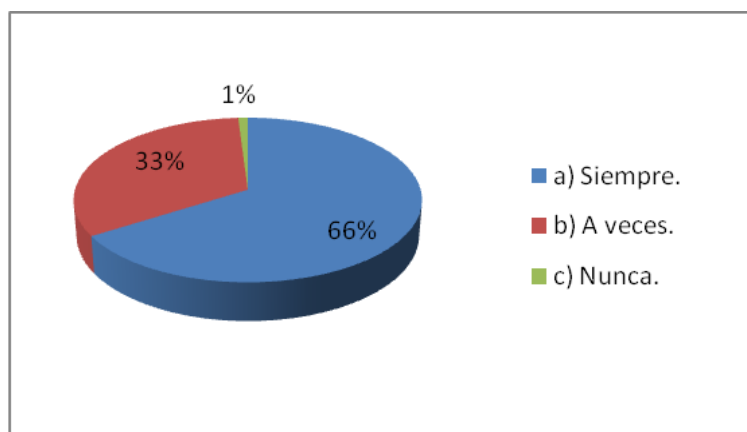


Gráfico N° 7: Uso de un software para procesar imágenes.

Síntesis Pregunta 7: La sociedad progresista se ha formado gracias al trabajo y la práctica entusiasta de los seres humanos, además de sus inquietudes. Los datos anteriores muestran favorablemente, que el 66% de los encuestados usan siempre un software para procesar imágenes digitales, apoyado por un 33% que lo usan a veces.

8. Si usted tuviera la oportunidad de utilizar un software para procesar imágenes debería ser:

Cuadro N° 9.0: Software para procesar imágenes digitales

SOFTWARE PARA PROCESAR IMÁGENES DIGITALES		
OPCIÓN	f	%
a) Fácil de usar.	16	16
b) Rápido.	16	16
c) Eficiente.	15	15
d) Entendible y eficiente.	53	53
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

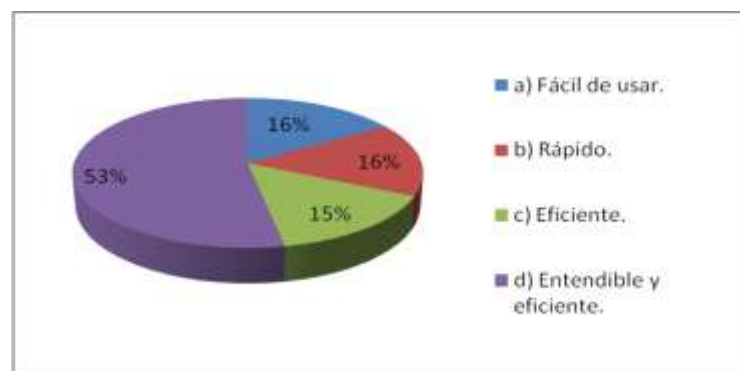


Gráfico N° 8: Software para procesar imágenes digitales.

Síntesis Pregunta 8: El procesar una imagen digital es una tarea que se la realiza de una forma sencilla, claro, dependiendo del software con el que se este trabajando, según los datos obtenidos por las encuestas planteadas en la carrera de Diseño Gráfico. El 53% dice desear un software para procesar imágenes, entendible y eficiente, precedido de un empate de 16% de las opciones: fácil de usar, y rápido.

9. Si usted adquiriera un manual para el uso de un determinado software le gustaría que sea:

Cuadro N° 10.0: Manual de uso

MANUAL DE USO		
OPCIÓN	f	%
a) Completo.	66	66
b) Básico.	6	6
c) Específico.	28	28
TOTAL	n = 100	100%

Elaborado por: Grupo Investigador

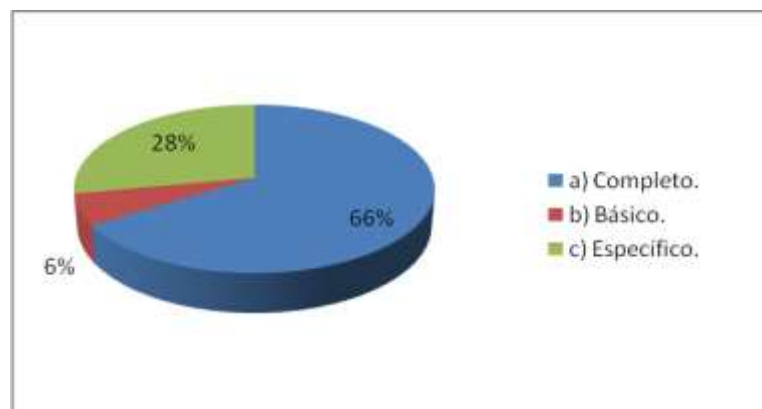


Gráfico N° 9: Manual de uso.

Síntesis Pregunta 9: Los datos, resultado de las encuestas planteadas a los docentes y estudiantes de la especialización en Diseño Gráfico; muestran que el 66% de los mencionados anteriormente, prefieren que el manual de uso de algún determinado software sea completo, seguido por un 28% que lo prefiere específico. Razón por la cual el manual del software prototipo, tema de esta tesis, será completo.

2.3.1 Verificación de la Hipótesis

La hipótesis planteada al inicio de la investigación es: La implementación de un software prototipo de procesamiento y extracción de rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial, ayudará a la especialización de Diseño Gráfico de C.I.Y.A. en la Universidad Técnica de Cotopaxi.

De acuerdo a la información de campo facilitada por los estudiantes y docentes de la especialización de Diseño Gráfico de C.I.Y.A., consideramos que la hipótesis ha sido verificada por lo siguiente:

Los datos más relevantes obtenidos por medio de las interrogantes realizadas a los estudiantes y docentes en la encuesta, mismos que contestan con un 88 por ciento a favor de una alta importancia del procesamiento de imágenes digitales dentro de la Especialización de Diseño Gráfico. Además que tienen claro la idea de que es procesar una imagen digital, ya que el 55 por ciento responden exactamente a este planteamiento. Deseando tener una nueva aplicación para procesar imágenes digital de características entendible y eficiente en un 53 por ciento y a su vez un favorable 66 por ciento de los encuestados, que desean un manual de usuario. Dando lugar a la creación de nuestro software prototipo que les ayude en cierto trabajo con imágenes en condiciones favorables y no de paquetes comerciales que a la hora de su funcionamiento disminuyen el rendimiento del computador y acreditan molestias y dificultades en su manejo.

Finalmente, el SOWMEIM v1.0 se inicia como una versión muy avanzada en su categoría, capaz de acondicionar la imagen sin mayor problema a la necesidad del usuario

2.4 Posibles funcionalidades, y ventajas para la solución de problemas con respecto a las imágenes en la especialización de Diseño Gráfico

El grupo investigador afirma que el papel principal que realiza el software acreditado por esta tesis; es el mejoramiento de una imagen degradada, es decir, al momento en que un usuario a través de su cámara obtuvo la captura o adquisición de una imagen algo distorsionada, el software prototipo le permitirá mejorar la apariencia visual de dicha imagen, teniendo así una herramienta automática para la solución de este problema, y así permitir al usuario trabajar con imágenes digitales satisfactoriamente. Pero más allá de esto existen múltiples aplicaciones que a largo plazo podría realizarse con el software prototipo, en especial en la especialización de diseño gráfico computarizado.

Gracias a la ayuda del libro de Domingo Mery, “Visión Artificial” y sus contenidos, menciona que:

2.4.1 Fotogrametría: En la fotogrametría se persigue realizar mediciones del espacio 3D a partir de fotografías tomadas de él. De esta manera es posible medir superficies, construcciones, objetos, etc. Asimismo se puede llevar a cabo una topología de un terreno.

2.4.2 Rectificación Métrica: Mediante esta técnica es posible hacer correcciones de perspectiva y correcciones de distorsión de lente.

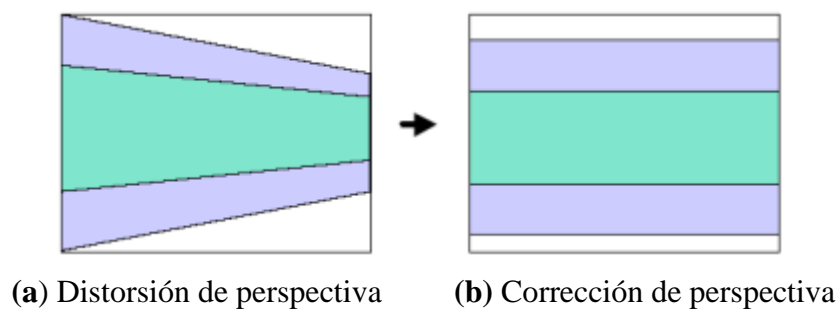


Figura 14.0: Ejemplo de rectificación de perspectiva

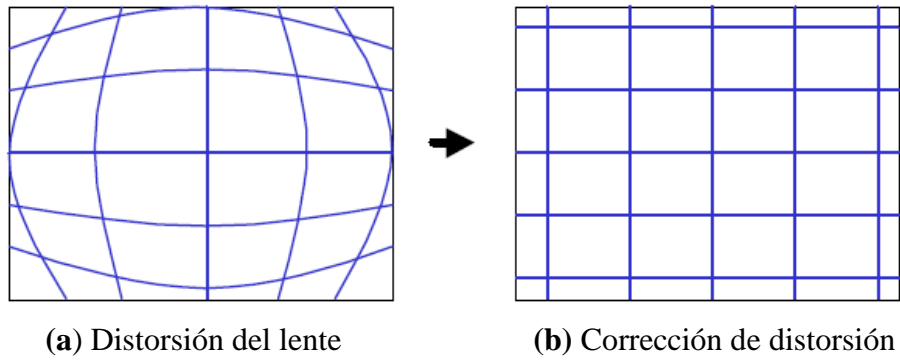


Figura 14.1: Ejemplo de rectificación de distorsión de lente

2.4.3 Reconstrucción 3D: A partir de las vistas, mediante la técnica de triangulación, es posible obtener un modelo 3D del objeto proyectado en las vistas. El principio de triangulación es mostrado en la Figura 14.2: sabiendo que los puntos A y B son proyecciones de un mismo punto tridimensional Q , es decir A y B son correspondientes, y conociendo los centros ópticos de la proyección C_1 y C_2 , se puede encontrar el punto Q a partir de la intersección entre las dos rectas $(C_1; A)$ y $(C_2; B)$.

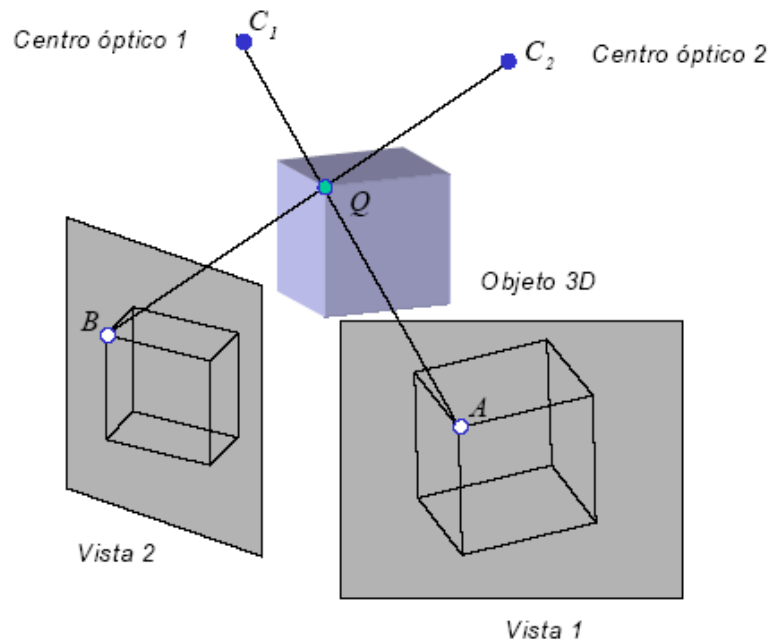


Figura 14.2: Triangulación: estimación de Q a partir de A y B

2.4.4 Matching y Tracking: Por medio del Matching y Tracking es posible encontrar la correspondencia entre puntos de varias imágenes. Los puntos

correspondientes son aquellos que representan una proyección del mismo punto físico en el espacio 3D. En la Figura 14.3 se puede apreciar tres vistas de una taza tomadas por una cámara fija mediante la rotación del eje central de la taza. Se puede observar que los puntos m_1 , m_2 y m_3 en las imágenes 1, 2 y 3 respectivamente, son correspondientes entre si porque son proyecciones del mismo punto m de la taza.

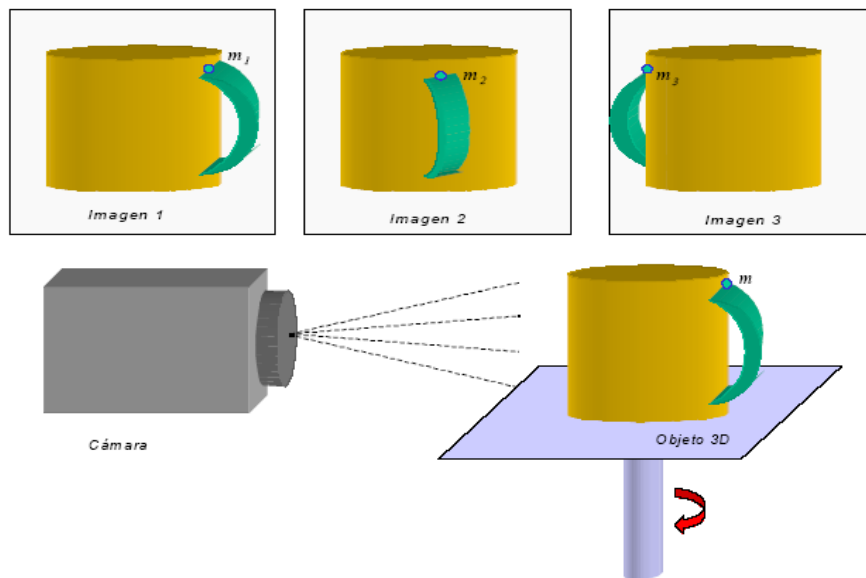


Figura 14.3: Correspondencia en tres puntos.

2.4.5 Computación Gráfica: Si se tiene un modelo de la formación de la imagen f : 3D a 2D, es posible entonces simular gráficamente las vistas bidimensionales que se obtendrían de un objeto tridimensional. Las aplicaciones de realidad virtual emplean esta teoría.

2.4.6 Estimación de Movimiento: Mediante una cámara que toma imágenes de un objeto en movimiento es posible estimar el movimiento del objeto a partir de los puntos de correspondencia en la secuencia de imágenes.

Según el grupo investigador el software prototipo para procesamiento de imágenes y extracción de rasgos, con técnicas de Inteligencia Artificial, será de gran ayuda para un estudiante de diseño gráfico cuando al trabajar con una imagen

en una computadora pueda cambiar el color de la imagen, eliminar el ruido, eliminar aspectos de la imagen no deseados, cambiar tonalidades, extraer características, entre otras cosas. Y a nivel de herramienta informática ser un software con un interfaz entendible, fácil de usar, con excelentes resultados en la tarea a desempeñar, y sobre todo rápido con respecto a los potentes programas que existen en el mercado informático para el procesamiento y tratado de imágenes.

2.5 Esquemas técnicos y figuras en el desarrollo del software prototipo para el Procesamiento de Imágenes y Extracción de Rasgos

2.5.1 Introducción: Al iniciar el desarrollo de un sistema en Inteligencia Artificial es imperioso conocer, lo que se quiere llegar a diseñar y las respectivas definiciones para poder hacerlo. A continuación en la Figura 15.0 mostramos, en parte, el camino de lo que realiza un sistema de visión por computadora.

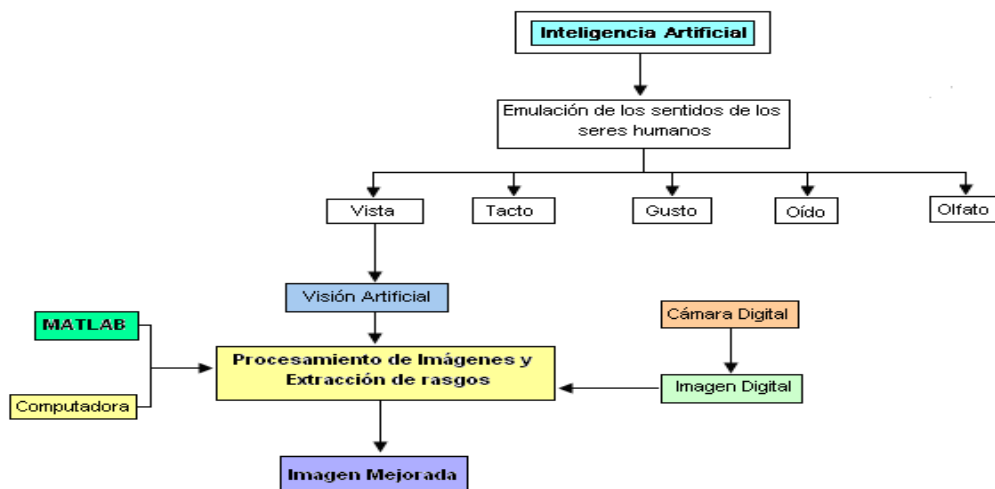


Figura 15.0: Esquema general de Inteligencia Artificial, y la emulación de la vista humana.

El libro: *Artificial intelligence: an engineering approach*, y en complicidad con el libro: *Inteligencia artificial* proporciona la siguiente información resumida en las siguientes líneas.

La Consecución del software prototipo para procesamiento de imágenes con técnicas en Inteligencia Artificial, conlleva un proceso de desarrollo iterativo controlado (controlled iterative process development), donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Primero se identifican los riesgos y después se prueba la aplicación para que éstos se hagan mínimos. Cuando la implementación pasa todas las pruebas que se determinan en el proceso, ésta se revisa y se añaden los elementos modificados.

La Figura 15.1, muestra el proceso de desarrollo iterativo controlado, y lo escrito en el párrafo anterior.

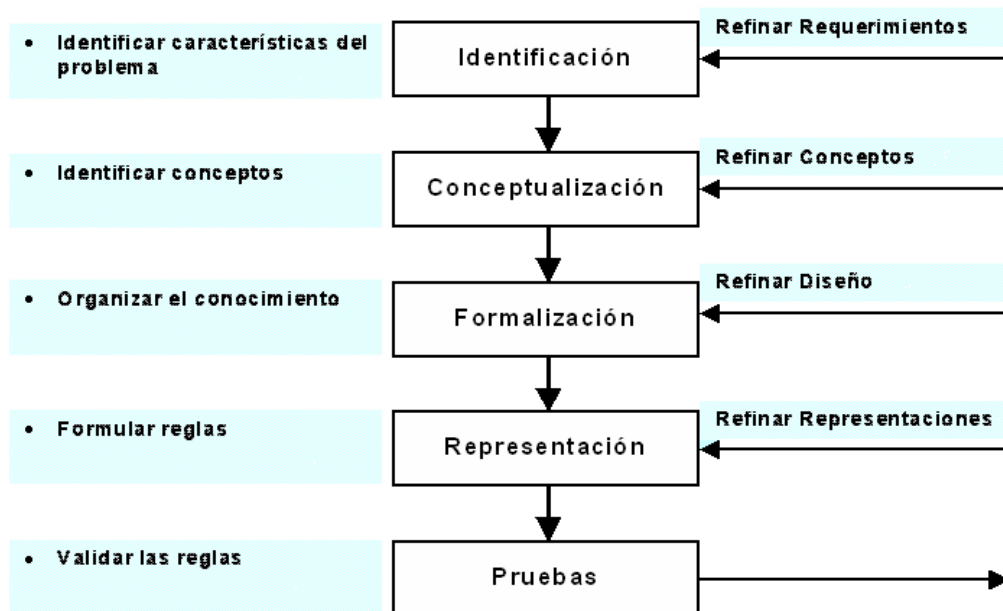


Figura 15.1: Proceso de desarrollo iterativo controlado.

Un sistema de visión por computadora se compone básicamente de los siguientes elementos:

Tarjeta de captura o adquisición de imágenes: es la interfaz entre el sensor y la computadora o módulo de proceso que permite al mismo disponer de la información capturada por el sensor de imagen.

Algoritmos de análisis de imagen: es la parte inteligente del sistema. Su misión consiste en aplicar las transformaciones necesarias y extracciones de información de las imágenes capturadas, con el fin de obtener los resultados para los que haya sido diseñado.

Computadora o módulo de proceso: es el sistema que analiza las imágenes recibidas por el sensor para extraer la información de interés en cada uno de los casos implementando y ejecutando los algoritmos diseñados para la obtención de los objetivos.

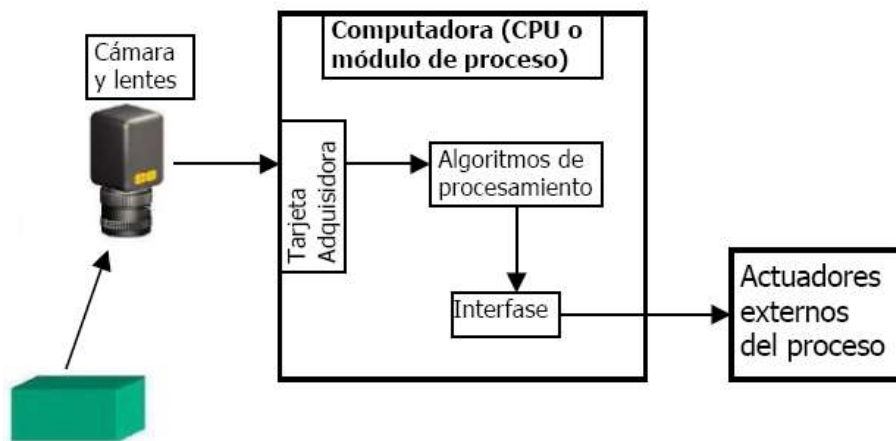


Figura 16.0: Elementos de un sistema de visión por computadora.

2.5.2 Pasos Fundamentales del Procesamiento Digital de Imágenes

Dividiremos por métodos en que la entrada y salida es una imagen, y métodos en los que la entrada es una imagen y la salida son datos. Las partes fundamentales del PDI son las siguientes. Después de adquirir la imagen, en un procesamiento no siempre se siguen todos los pasos:

1. Adquisición de imágenes.- Generalmente incluye pre-procesamiento.

2. Mejora de la imagen.- La idea de este paso es obtener detalles que no se veían, o simplemente subrayar ciertas características de interés. Que se “vea mejor”.

3. Restauración de la imagen.- También mejora la apariencia de la imagen, a diferencia de la mejora de la imagen, subjetiva, la restauración es objetiva, en el sentido en que las técnicas de restauración tienden a ser modelos probabilísticos o matemáticos de degradación de la imagen (¿Cómo era esta imagen antes de ser dañada?).

4. Procesamiento del color.- Procesamientos especiales para el color.

5. Ondeletas.- Fundamentalmente utilizadas para representar imágenes en varios grados de resolución. Se utiliza principalmente en compresión.

6. Compresión.- Reduce el almacenamiento requerido para guardar una imagen, o el ancho de banda para transmitirla.

7. Procesamiento morfológico.- Herramientas para extraer componentes de la imagen útiles para la representación y descripción de formas.

8. Segmentación.- Divide una imagen en sus partes constituyentes.

9. Representación y descripción.- Se toman decisiones tales como si la forma obtenida debe ser tratada como un frontera o una región, y extrae atributos que resultan en información cuantitativa de interés.

Base de conocimiento.- Nos sirve de almacenamiento de información: Ya sean los datos de las imágenes en cada paso, o heurísticas que nos ayudarán a obtener mejores imágenes en subsecuentes procesamientos.

Según el grupo investigador acoge estas conceptualizaciones como apoyo indispensable para el diseño de la interfaz del software prototipo, ya que gran parte de información concuerda con lo investigado y estudiado.



Figura 16.1: Pasos del PDI

2.5.3 Componentes de un sistema de PDI de propósito general

1. **Sensor.-** Es un aparato físico sensible a la energía radiada por el objeto.

2. **Hardware de procesamiento de imágenes especializado.-** Usualmente consta de un digitalizador que convierte la salida del sensor a formato digital, y de hardware que realiza otras operaciones, como una ALU, que procesa operaciones en paralelo de imágenes completas. Su característica más distintiva es la velocidad (realiza operaciones que requieren salidas de datos muy rápidas).

3. **Computadora.-** En un sistema de propósito general puede ser desde una PC hasta una supercomputadora.

4. **Software.-** Módulos especializados que llevan a cabo tareas específicas.

5. **Almacenamiento en masa.-** Obligatorio en aplicaciones de procesamiento de imágenes. El almacenamiento se divide en tres categorías:

- A corto plazo, utilizado durante el procesamiento (ej. Memoria de la computadora, o memorias especiales “frame buffers”, éstas últimas se utilizan en ZOOMS digitales).
- En línea, para carga relativamente rápida (discos magnéticos u ópticos).
- De archivo, de acceso poco frecuente (cintas magnéticas y discos ópticos).

6. Dispositivos de despliegue de imágenes.- Por ejemplo, monitores a color (de preferencia planos, para evitar distorsiones).

7. Dispositivos de Impresión (Hardcopy).- Impresoras láser o tinta, cámaras de película, dispositivos sensibles al calor, unidades digitales como CD-ROM.

8. Red.- Casi una función por defecto en cualquier sistema computacional moderno, lo más importante es el ancho de banda.

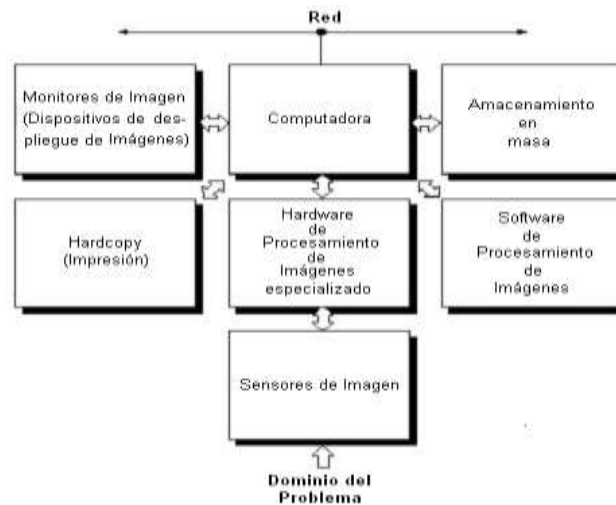


Figura 16.2: Componentes de un sistema de PDI de propósito general

Que importante la participación de Gonzáles R. y Wood R, al participar con los conceptos, estructuras y diagramas de bloque que hace todavía más captable el gran tema sobre procesamiento de imágenes.

“Visión es saber que hay y donde mediante la vista”

Aristóteles.

CAPITULO III

“FACTIBILIDAD EN LA IMPLEMENTACIÓN DE UN SOFTWARE PROTOTIPO DE PROCESAMIENTO Y EXTRACCIÓN DE RASGOS, PARA MEJORAR LA CALIDAD DE LA IMAGEN; UTILIZANDO TÉCNICAS DE INTELIGENCIA ARTIFICIAL, PARA LA ESPECIALIZACIÓN DE DISEÑO GRÁFICO DE C.I.Y.A.”

3.1 Presentación

Para este apartado el libro de Nixon M. y Aguado A. dice un preámbulo, que a su vez es una reseña histórica sobre el procesamiento de imágenes diciendo que:

En (1920) Bartlane cable Picture transmission. Envío de imágenes por cable entre Londres y N.Y. Se mejoró la calidad de las imágenes digitalizadas de los periódicos, enviadas por cable submarino entre Londres y Nueva York.

En (1964) la NASA, procesa por primera vez la imagen de la superficie de la luna desde la nave aéreo espacial Ranger7, el 31dejulio a las 9:09 A.M.

En (1970) el CT SCAN inventado por G.N. Hounsfield y A.M. Cormack (Nobel, 1979, en medicina) basado en Rayos X por W.C. Roentgen (Nobel, 1901, en física). Se procesó por primera vez una imagen de rayos X.

Pero sin duda es en el Siglo 21, cuando tenemos la implacable ola de la "Revolución de la Información". La denominada "Autopista de la Información" ha llegado ya a nuestras casas y está conectada a nuestros televisores y PCs de alto rendimiento. El banco en casa, compra por internet, CDs de audio, DVDs... Se entregan ya en nuestras casas y nos permiten el placer de ver los programas que queramos con confort. Los teléfonos de imágenes visuales nos permiten ya un tipo de comunicación que hasta ahora había sido sólo posible con un, cara a cara con nuestro interlocutor.

También trabajar en casa en vez de desplazarse y trabajar en el lugar de trabajo se convertirá seguramente en estos años venideros en algo muy usual.

La información consta de incontables elementos, incluidas letras, figuras, sonidos, voces, imágenes visuales e imágenes audio-visuales. En la "Revolución de la Información", toda esta información se guardará y transferirá a través de una tecnología que combina productos multimedia, y técnicas de digitalización. De hecho, la competitividad internacional dependerá inevitablemente de un desarrollo que combine las fuerzas del hardware (red de comunicación super-rápida) y del software (capacidad de la red de información ultra-eficiente).

En esta era multimedia, el procesamiento de imágenes digitales es, sin lugar a dudas, la tecnología más importante que nos puede asistir en la producción de imágenes.

Los seres humanos confían fundamentalmente en el sentido de la vista. Recogemos 99 % de nuestra información a través de nuestros ojos. El procesamiento de imágenes es un proceso que concierne tanto a ordenadores como a algoritmos diversos para un propósito específico, una vez que hemos adquirido imágenes de interés a través de una cámara o escáner.

La avidez y la potencialidad, por imitar tecnológicamente el mundo que los rodea, se hace cada vez algo tan cercano y ya no un sueño imposible, y todas estas destrezas tiene que ser debidamente encausada y aprovechadas para comodidad de los seres humanos y el planeta.

El conocimiento de esta realidad ha provocado proponer, el desarrollo y posterior implementación del software prototipo de procesamiento y extracción de rasgos; utilizando técnicas de Inteligencia Artificial, el que hoy estamos presentando a fin de incursionar en el mejoramiento de imágenes digitales; para que en complicidad con un eficiente computador, sea motivo de una percepción visual real excelente, en los alumnos de la especialización de Diseño Gráfico y en el usuario del software prototipo. Siendo así parte de la solución y no del problema.

3.2 Justificación

La evolución tecnológica, como la automatización, son ítems que globalizan al mundo, a las universidades y por ende a los estudiantes, a la consecución de nuevas metas y objetivos en el desarrollo de sus carreras, además de ser bases con las cuales se genera la dependencia de la investigación y la ciencia.

La elaboración de este proyecto de Tesis cuenta con el aval y la ayuda de un director y suficiente material de apoyo que nos permite obtener varios criterios de información que se utilizados para analizar, diseñar, desarrollar, e implementar el software prototipo.

El procesamiento de imágenes y extracción de rasgos; con técnicas de Inteligencia Artificial; nos permitirá manipular imágenes digitales en un computador con el fin de obtener información objetiva de la escena captada por una cámara. Para que estudiantes de Diseño Gráfico puedan manipular imágenes a su agrado, y en especial el mejoramiento de la calidad de las imágenes. Las computadoras que tenemos hoy en día sobre el escritorio, con un estupendo software; pueden realizar

los cálculos necesarios para el procesamiento de imágenes, siempre y cuando estén adoptados por el usuario; y las aplicaciones potenciales son muchas, por ejemplo:

- Interacción hombre máquina más natural mediante el uso de la información proveniente en este caso de una cámara.
- Captura de movimiento.
- Seguridad.
- Gestos y animación por computadora.

La herramienta y lenguaje de programación que utilizamos, es MATLAB, permitiendo de alguna manera mejorar la utilidad que se le da en la carrera ha este potente y muy avanzado programa, que abarca desde fórmulas matemáticas, métodos numéricos, algoritmos, hasta librerías eficientes en el desarrollo de simulaciones, realidad virtual, y redes neuronales.

La finalidad de este proyecto es introducir un software prototipo, como nueva técnica en el tratamiento de imágenes digitales, para el área de Diseño Gráfico, utilizando un lenguaje de programación orientado a objetos y sobre todo eficiente.

De hecho no solo se pretende cubrir necesidades al momento de mejorar y trabajar con una imagen, sino que, presentar diversas posibilidades para posteriores versiones de la aplicación en el ámbito académico, como también generar alternativas en el desarrollo de proyectos en nuestra carrera.

3.3 Objetivos

Existen diversidad de objetivos que pueden brotar a un material investigativo y técnico de Inteligencia Artificial, pero la delimitación de este tema a una mencionada tarea permite plantearnos objetivos tanto general como específicos.

El software prototipo de Procesamiento y Extracción de Rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial pretende lograr los siguientes objetivos:

3.3.1 *Objetivo del Software prototipo SOWMEIM v1.0:*

- ☑ Implementar un software prototipo de Procesamiento y Extracción de Rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial, para la especialización de Diseño Gráfico de C.I.Y.A., en el periodo noviembre 2007 a julio 2008.

3.3.2 *Objetivos específicos del Software prototipo SOWMEIM v1.0:*

- ✓ Crear un interfaz adecuado en el software prototipo para el procesamiento de imágenes y extracción de rasgos.
- ✓ Optimizar las etapas de construcción con algoritmos factibles en el software prototipo, para el procesamiento de imágenes.
- ✓ Implementar una herramienta eficiente para el tratamiento de imágenes degradadas.
- ✓ Obtener una visualización adecuada de los rasgos de la imagen.
- ✓ Realizar una guía de usuario para el manejo del software prototipo.

3.3.3 *Objetivos Conceptuales:*

- ✓ Conocer las principales técnicas para el procesamiento y análisis de imágenes.
- ✓ Analizar, comparar e implementar en la computadora las diferentes técnicas, aplicándolas al mejoramiento de imágenes.

3.3.4 *Objetivos Procedimentales:*

- ✓ Analizar críticamente información actualizada y relevante.
- ✓ Seleccionar adecuadamente diferentes alternativas de solución de problemas.
- ✓ Buscar y analizar información por nuestra cuenta, y ser capaces de aplicarlas a problemas reales.

- ✓ Utilizar la caja de herramientas para el procesamiento de imágenes que posee MATLAB, a nuestra conveniencia.

3.4 Factibilidad de aplicar el software prototipo, en la especialización de Diseño Gráfico

3.4.1 Introducción: Según la página web www.monografias.com/imagen/pixel/ una imagen digital se compone de píxeles, que se pueden imaginar como puntos de la pantalla. Una imagen digital es una instrucción de como colorear cada píxel. Una imagen típica puede ser de 512×512 píxeles. En el caso general una imagen es de resolución $m \times n$ si se compone de m píxeles en la dirección vertical y n píxeles en la horizontal. Supongamos que tenemos una imagen de resolución 512×1024 . Esto significa que los datos de la imagen deben contener información de $512 \times 1024 = 524288$ píxeles, que requiere una gran cantidad de memoria. Por tanto, la compresión de la imagen es fundamental para un procesamiento eficiente de la misma.

Según el grupo investigador deduce que en un área en donde la raíz de toda su estructura son las imágenes, como es el caso del diseño gráfico; es indispensable tener como herramienta de trabajo un software, que permita procesar mencionadas imágenes y que el usuario pueda interactuar con su imagen en las características y condiciones que lo requiera. Por ejemplo: para el mejoramiento de la imagen es necesario eliminar el ruido.

Según la página de internet [http://www.aitex.es/estudios/Vision Artificial1.pdf](http://www.aitex.es/estudios/Vision%20Artificial1.pdf). define los siguientes enunciados:

3.4.2 Ruido en imágenes: Todas las imágenes tienen cierta cantidad de ruido, la cual se puede deber a la cámara o al medio de transmisión de la señal. Generalmente el ruido se manifiesta como píxeles aislados que toman un nivel de gris diferente al de sus vecinos. Los algoritmos de filtrado, permiten eliminar o disminuir este ruido. El ruido puede clasificarse en los siguientes tipos:

3.4.2.1 Gaussiano: produce pequeñas variaciones en la imagen; generalmente se debe a diferentes ganancias en la cámara, ruido en los digitalizadores, perturbaciones en la transmisión, etc.

3.4.2.2 Impulsional (sal y pimienta): el valor que toma el píxel no tiene relación con el valor ideal, sino con el valor del ruido que toma valores muy altos o bajos (puntos blancos y/o negros) causados por una saturación del sensor o por un valor mínimo captado, si se ha perdido la señal en ese punto. Se encuentran también al trabajar con objetos a altas temperaturas, ya que las cámaras tienen una ganancia en el infrarrojo que no es detectable por el ojo humano; por ello las partes más calientes de un objeto pueden llegar a saturar un píxel.

3.4.2.3 Multiplicativo: La imagen obtenida es el resultado de la multiplicación de dos señales.

También para mejorar una imagen puede ser necesario transformarle a otro formato. Si tenemos una imagen en formato JPEG, la podemos leer en MATLAB. Sin embargo, para que la podamos tratar, debemos pasarla a otro formato. Aquí explicamos cuatro maneras distintas.

3.4.3 Escala de grises.- Esta es una de las formas más habituales. Una imagen se representa como una matriz donde cada elemento tiene un valor correspondiente al brillo/oscuridad de cada píxel. Existen dos formas para representar el número que indica este brillo. La primera es con el tipo de dato double. Asigna un número en coma flotante entre 0 y 1 a cada píxel. El valor 0 corresponde al negro y el valor 1 al blanco. La segunda forma usa el tipo de dato unit8, y asigna un entero entre 0 y 255 para representar el brillo de cada píxel. El valor 0 corresponde al negro y el valor 255 al blanco.

3.4.4 Imagen binaria.- Este formato almacena una imagen como una matriz, pero solamente puede colorear un píxel como blanco o negro, sin valores intermedios. Asigna el valor 0 para negro y 1 para el blanco.

3.4.5 Imagen indexada.- Esta es una forma práctica de representar imágenes en color. Una imagen indexada almacena una imagen como dos matrices.

La primera matriz tiene el mismo tamaño que la imagen y un número para cada píxel. La segunda matriz se denomina mapa de color y su tamaño puede diferir del de la imagen. Los números de la primera matriz son una instrucción de qué color usar según el mapa de color.

3.4.6 Imagen RGB.- Esta es otra forma para imágenes en color. Una imagen se representa como tres matrices, cada una del mismo tamaño que la imagen original. La primera matriz corresponde al color rojo, la segunda al color verde y la tercera al color azul, e indican la porción de cada color que tiene un píxel.

3.4.7 Imagen multi-cuadro.- En algunas aplicaciones queremos estudiar una sucesión de imágenes. Esto es muy común en biología y medicina donde se quiere estudiar una sucesión de cortes de una célula. Para estos casos, el formato multi-cuadro es una forma conveniente de trabajar con una sucesión de imágenes.

Según el grupo investigador otro inconveniente consultado que puede afectar a una imagen es, el efecto *alias* que se presenta al reducir el tamaño de una imagen. Esto es debido a que se refleja una pérdida de información cuando se reduce el tamaño de una imagen.

3.5 Impacto y vida útil del software prototipo

El grupo investigador aclara que el software prototipo SOWMEIM v1.0, es una herramienta, nacida del programa MATLAB y sus “Toolboxes” o caja de herramientas, con el cual un usuario puede mejorar la calidad de la imagen obtenida por la cámara digital conectada a su ordenador, este es el principal propósito con el cual fue diseñado SOWMEIM v1.0; y para que esta corrección de errores se produzca de una manera primordial, hemos incorporado en la herramienta, diferentes tipos de filtros que es la estrategia más importante para propiciar en la imagen final una mejor percepción visual.

Además de tener un interfaz sencilla de usar, el software SOWMEIM v1.0 posee un manual de usuario que permitirá al estudiante realizar un correcto procesamiento y mejoramiento de la imagen; claro siempre y cuando se le de un buen uso al software.

Para nuestro criterio la el software SOWMEIM v1.0 tendrá un excelente impacto en aquellas personas que requieran de una herramienta rápida y eficiente en su función, entendible y que no ocupe mucho espacio de memoria. Además de todos aquellos estudiantes de diseño gráfico que les guste su carrera y que amen trabajar con imágenes en distintas formas y formatos. Finalmente también servirá como ayuda para estudiantes de ingeniería en informática y sistemas que vean de la Inteligencia Artificial y la Visión Artificial, uno de los temas para concluir su carrera.

La vida útil de SOWMEIM v1.0 por obvias razones, está determinado por el avance tecnológico (nuevos lenguajes de programación o mejoras de los existentes), y por las posteriores versiones que se obtenga de este software prototipo.

El mundo evoluciona día a día, tanto en el aspecto de software y hardware, por tal motivo no será ajeno encontrar un programa o una página de Internet que cumplan con similares tareas que el SOWMEIM v1.0 y en mejores condiciones; dejando en claro, que la vida útil de esta herramienta será como máximo de unos dos a tres años (en procesamiento de imágenes), y como mínimo de un año y medio (en visión artificial y mejoramiento de imágenes).

3.6 Desarrollo del software prototipo

El grupo investigador plantea que una vez preparada la escena para ser capturada por la cámara de Visión Artificial, se deberá de tratar el aspecto tanto de la cuantificación como de la codificación que sufre la imagen al ser introducida en el computador. Depositada la información, como una matriz o conjunto de matrices

de valores discretos se procederá, a un procesado de la imagen y tratar de obtener una nueva imagen que mejore su calidad o bien destaque algún atributo primario de los objetos capturados. En el aspecto de la calidad tratará de corregir las posibles faltas de iluminación, la eliminación del ruido o de aumentar el contraste en la imagen. Mientras en el realce, su objetivo es destacar los bordes de los objetos, regularizar sus colores, acentuar sus texturas, etc.

Las técnicas de procesamiento de las imágenes pueden ser clasificadas en dos grandes ramas: a) las procedentes del procesamiento de señales y b) aquellas que son de carácter heurístico y que nacieron para mejorar algún aspecto primario de la imagen. Por consiguiente el sistema de Visión por Computadora que maneja el software prototipo SOWMEIM v1.0 está estructurado en la siguiente forma:

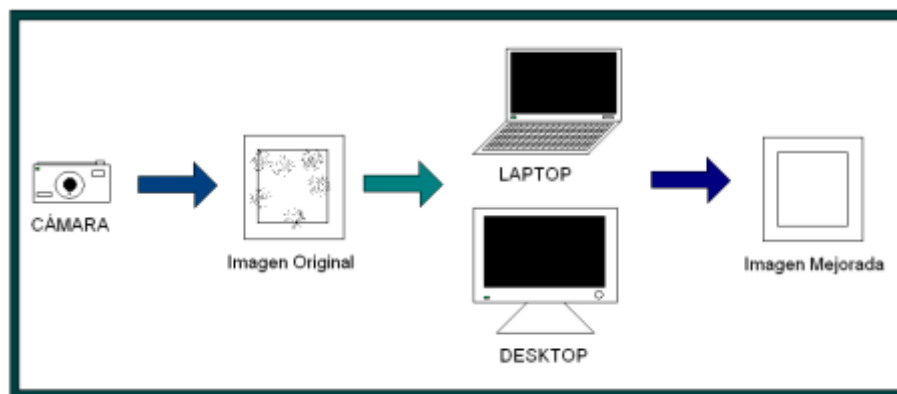


Figura 17.0: Visión por Computadora del software prototipo SOWMEIM v1.0

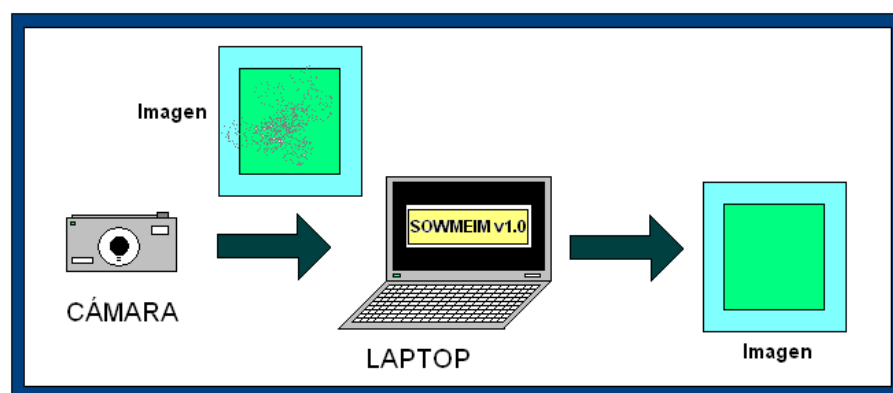


Figura 17.1: Funcionalidad del software SOWMEIM v1.0

El software SOWMEIM v1.0 está basado en las aplicaciones del software MATLAB 7.0 apoyado con sus herramientas de adquisición y procesamiento de imágenes; y tiene la siguiente interfaz:

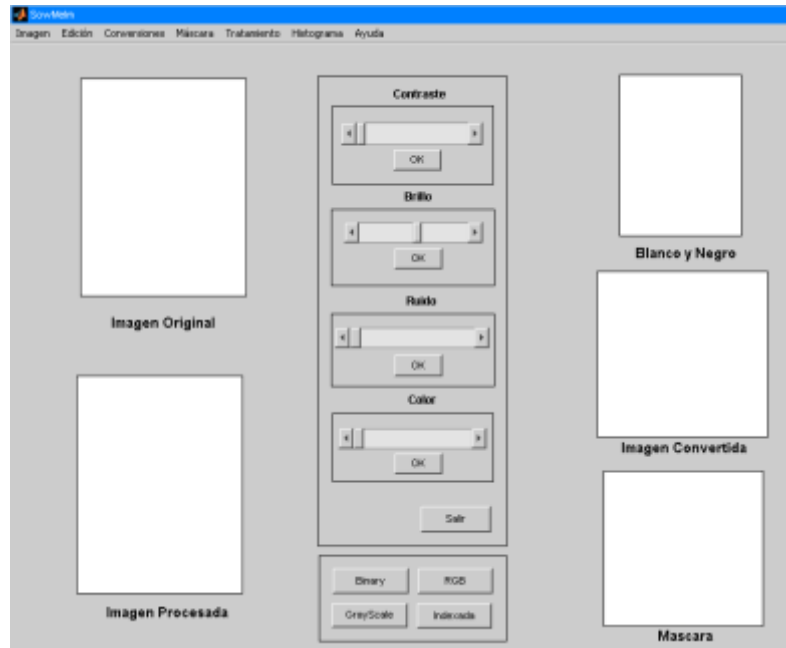


Figura 17.2: Interfaz del software prototipo SOWMEIM v1.0

3.6.1 *Procesamiento de imágenes*

El termino procesamiento de imágenes, se refiere a la manipulación de imágenes digitales por medio de una computadora; existe tres tipos de procesamiento:

a) Procesamiento de bajo nivel (Low-level processing): Procesos caracterizados porque las entradas y salidas son imágenes. Operaciones básicas como reducción de ruido, mejoramiento del contraste.



Figura 18.0: Procesamiento de bajo nivel

b) Procesamiento de nivel medio (Mid-level processing): Implica tareas tales como segmentación (particionar una imagen en regiones u objetos), descripción de objetos para reducirlos a una forma tal que se puedan procesar y clasificar los objetos individuales (reconocimiento).

Generalmente las entradas son imágenes y la salida **atributos** extraídos de las mismas (por ej. Bordes, contornos, o la identificación de los objetos).



Figura 18.1: Procesamiento de nivel medio

c) Procesamiento de alto nivel (High-level processing): Funciones asociadas a visión (comprender la imagen), es decir, implica descubrir el sentido de una imagen, como una conclusión más general, un análisis de la escena.

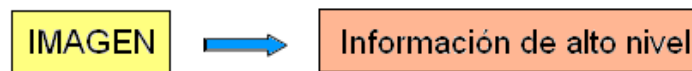


Figura 18.2: Procesamiento de alto nivel

3.6.1.1 Procesamientos básicos de imágenes

MATLAB lee, escribe y despliega data de imágenes de byte y de punto flotante. La herramienta de Procesamiento de Imágenes construye en este fundamento y en dos de las fortalezas de MATLAB: La estructura de matriz y la visualización de datos, que lo hace un ambiente natural para operar y desplegar imágenes y señales. Las operaciones de Procesamiento de Imágenes incluyen:

- Operaciones morfológicas binarias de imágenes.
- Reducción de ruido y realzamiento de imagen.
- Operaciones geométricas, etc.
- Remover defectos.
- Remover problemas por movimiento o desenfoque.

- Mejorar ciertas propiedades como color, contraste, estructura, etc.

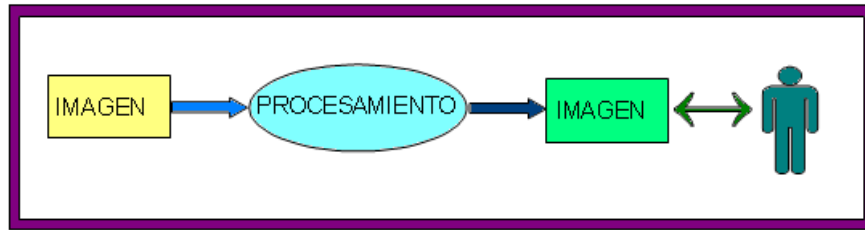


Figura 19.0: Diagrama en bloques del procesamiento de imágenes.

Ejemplos de procesamientos

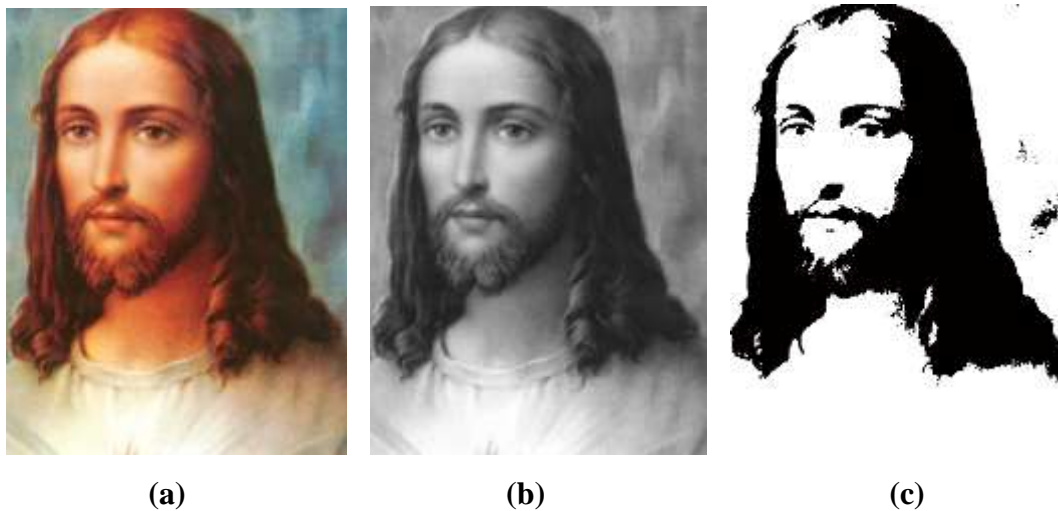


Figura 19.1: Imagen de Jesucristo a) Imagen RGB, b) Imagen Escala grises, c) Imagen en Blanco y Negro.

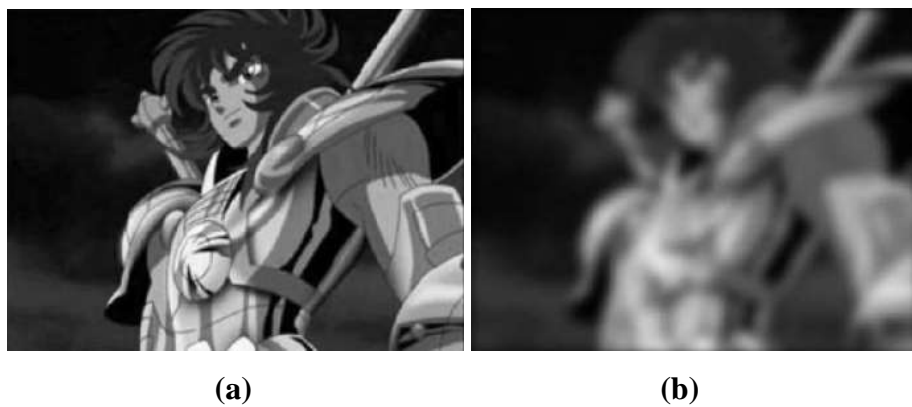


Figura 19.2: Imagen de una caricatura a) Imagen original, b) Imagen aplicada un filtro Pasa bajas.



(a)



(b)

Figura 19.3: a) Imagen aplicada un filtro Pasa altas, b) Imagen aplicada un filtro Pasa bandas.



(a)



(b)

Figura 19.4: Imagen de la biblioteca de MATLAB a) Imagen original, b) Imagen binarizada.



(a)



(b)

Figura 19.5: Imagen de la biblioteca de MATLAB a) Imagen original, b) Imagen aumentada el brillo.

3.6.1.2 Extracción de rasgos

Algunas características geométricas que se usan comúnmente en la extracción de rasgos son; como primer punto la extracción de bordes, y en segundo punto la segmentación.

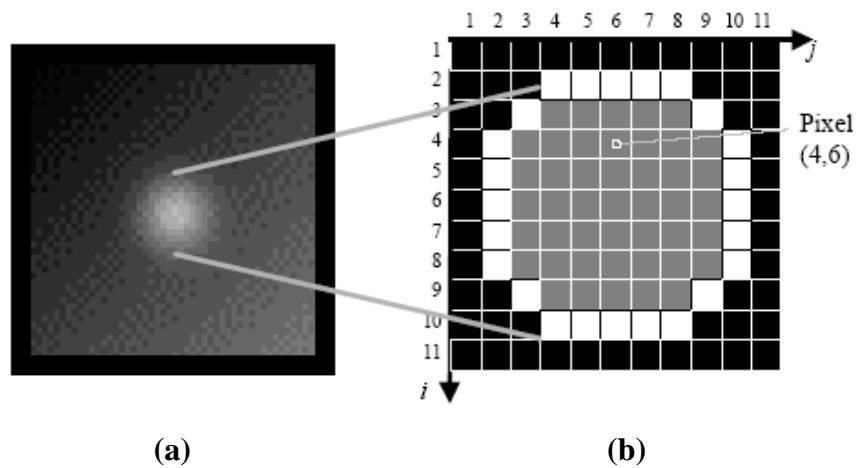


Figura 20.0: Ejemplo de una región: a) Imagen. b) Región segmentada

Ejemplo de extracción de bordes y de segmentación

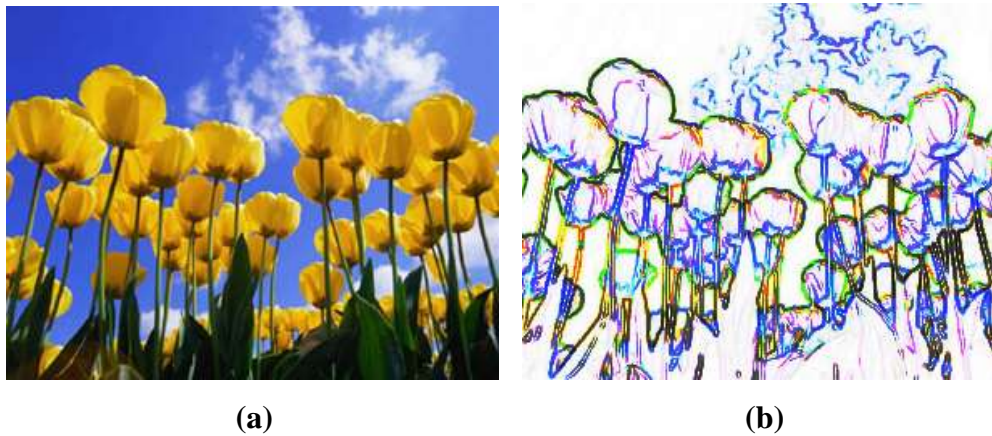


Figura 20.1: a) Imagen original. b) Imagen en bordes.



(a)



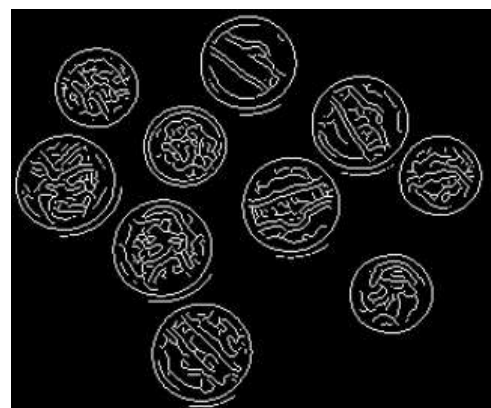
(b)

Figura 20.2: a) Imagen original. b) Segmento de la imagen.

Ejemplo de extracción de rasgos



(a)



(b)



(c)



(d)

Figura 20.3: a) Imagen original. b) Bordes de la imagen. c) Segmento de la imagen. d) Rasgos de la imagen.

3.6.1.3 Información final sobre Visión Artificial

Como datos preliminares recordamos que el programa MATLAB y los Toolboxes poseen en sus librerías las funciones: *Adquisición de imágenes* (Exportar e importar imágenes y vídeo, Compatible con varios fabricantes), *Procesamiento de imágenes* (Procesamiento y análisis), adicionalmente *Simulink*; posee los Blockset de Procesamiento de Imágenes.

Teniendo en cuenta que la Visión Artificial es una disciplina que tiene como finalidad, la extracción de información del mundo físico a partir de imágenes, utilizando para ello un computador. Es indispensable mencionar, que se ha cumplido con los objetivos planteados por esta tesis.

El sistema de Visión Artificial desarrollado en este proyecto actúa sobre una representación de una realidad que le proporciona información sobre brillo, color, formas, etc. Estas representaciones suelen estar en forma de imágenes estáticas, escenas reales o imágenes en movimiento.

Los dispositivos más conocidos para capturar imágenes suelen ser: Cámaras fotográficas, Cámaras de televisión, Digitalizadores, Sensores de rango (láser), Sensores de ultrasonido, Rayos X (tomografía), Resonancia magnética.



(a)



(b)



(c)



(d)

Figura 21.0: Dispositivos para captura de imágenes. a) Frame grabber. b) Web-Cam. c) Cámara de seguridad. d) Cámara digital.

Gran cantidad de expertos en Visión Artificial; está de acuerdo que los pasos globales en un sistema de Visión por computadora son:

1. Generación de imagen por la cámara.
2. Evaluación y procesamiento de la imagen.
3. Transmisión y presentación de los resultados.

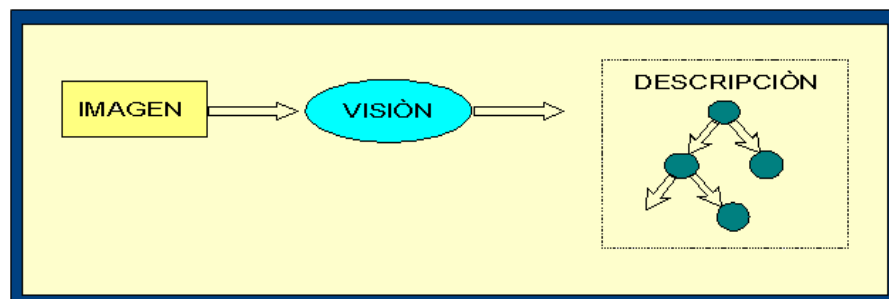


Figura 21.1: Proceso principal en Visión por computadora.

3.7 Elaboración del Manual para el Usuario

UNIVERSIDAD TÉCNICA DE COTOPAXI

CARRERA DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

INGENIERÍA EN INFORMÁTICA Y SISTEMAS COMPUTACIONALES

TEMA:

MANUAL DE USUARIO.

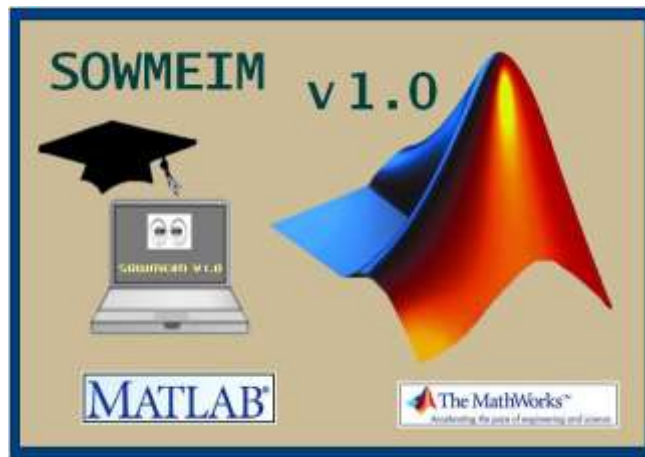


Figura 22.0: Carátula del manual del usuario.

SOFTWARE PROTOTIPO SOWMEIM v 1.0

Autores:

Chicaiza Toapanta Daniel Darío.

Falconí Hidalgo Renato Danilo.

Director:

Ing. Franklin Javier Montaluisa.

LATACUNGA - ECUADOR

PRÓLOGO

Éste software es una herramienta de procesamiento de imágenes y extracción de rasgos, en el que se emplea las utilidades de MATLAB en distintos algoritmos de programación, útiles para el mejoramiento de imágenes; y que promueve uno de los usos que tiene la Inteligencia Artificial, en su rama de la Visión por computadora.

Uno de los aspectos factibles del presente manual es que en los siguientes párrafos mencionamos algunos comandos básicos utilizados en el desarrollo del software prototipo SOWMEIM v1.0; recomendados por la mayoría de referencias bibliográficas. El SOWMEIM v1.0 prácticamente es un software que permitirá al usuario inicialmente, capturar una imagen desde una cámara digital o abrir una ya existente; y corregir ciertos errores como: disminución de ruido; aumento o disminución de contraste, brillo, color entre otros procesos que dan una fea percepción visual.

El interfaz del SOWMEIM v1.0 refleja una adecuada perspectiva para la manipulación de imágenes, tal es el caso; que puede ser comparado al menú obtenido en la pantalla del televisor cuando queremos configurar las propiedades del video. Motivo por el cual no presentará mayor dificultad en el uso esencial de esta herramienta.

Daniel Chicaiza – Renato Falconí

PRESENTACIÓN

En esta era multimedia, el procesamiento de imágenes digitales es, sin lugar a dudas, la tecnología más importante que nos puede asistir en la producción de imágenes.

Los seres humanos confían fundamentalmente en el sentido de la vista. Recogemos 99 % de nuestra información a través de nuestros ojos. El procesamiento de imágenes es un proceso que concierne tanto a ordenadores como a algoritmos diversos para un propósito específico, una vez que hemos adquirido imágenes de interés a través de una cámara o escáner.

La avidez y la potencialidad, por imitar tecnológicamente el mundo que los rodea, se hace cada vez algo tan cercano y ya no un sueño imposible, y todas estas destrezas tiene que ser debidamente encausada y aprovechadas para comodidad de los seres humanos y el planeta.

El conocimiento de esta realidad ha provocado proponer, el desarrollo y posterior implementación del software prototipo de procesamiento y extracción de rasgos; utilizando técnicas de Inteligencia Artificial, el que hoy estamos presentando a fin de incursionar en el mejoramiento de imágenes digitales; para que en complicidad con un eficiente computador, sea motivo de una percepción visual real excelente, en los alumnos de la especialización de Diseño Gráfico y en el usuario del software prototipo. Siendo así parte de la solución y no del problema.

Daniel Chicaiza – Renato Falconí

OBJETIVOS DEL MANUAL

Este Manual de Usuario perteneciente al software prototipo SOWMEIM v1.0, pretende lograr los siguientes objetivos:

- Permitir en el usuario una mejor manipulación del software.
- Ayudar al usuario a guiarse en los procesos que conllevan al mejoramiento de la imagen a procesar.
- Fomentar el uso correcto de técnicas de procesamiento de imágenes.
- Instruir en el lector del manual, el funcionamiento de cada botón, menú y opciones presentes en el software.
- Asociar al usuario con la imagen original y la imagen que quiere obtener.
- Estimular el desarrollo de habilidades a la hora de trabajar con imágenes.
- Favorecer en la toma de decisiones referentes a la percepción visual de la imagen.
- Despertar la interacción usuario - máquina.
- Permitir la motivación en el uso de software y aplicaciones que ocupan menor espacio de memoria en el PC.
- Ampliar el repertorio de herramientas útiles en el campo de Visión Artificial.

CÓMO USAR ESTE MANUAL

El Manual de usuario del software SOWMEIM v1.0, está estructurado una breve información teórica, prontas recomendaciones, seguido por párrafos en los que se señala la utilidad de cada menú, botón, y opción, todo en conjunto con una serie de figuras para hacer más ilustrativo al manual.

En las últimas líneas se encuentran pequeñas cápsulas teóricas interesantes para el entendimiento sobre el tema: procesamiento de imágenes y Visión por computadora. Las explicaciones se las ha redactado con un lenguaje sencillo, apropiado y comprensible.

RECOMENDACIONES PARA SU USO

Se debe tomar muy en cuenta las siguientes recomendaciones para efectuar el uso correcto de este manual y el software:

- ① Leer detenidamente los contenidos y recomendaciones.
- ② Evitar el mal uso del software.
- ③ Tener muy en claro los formatos de imágenes con los que puede trabajar el software SOWMEIM v1.0.
- ④ Capturar o cargar la imagen al software adecuadamente.

Software SOWMEIM v1.0

(Manual de Usuario)

Introducción

Este manual de usuario para el software prototipo SOWMEIM v1.0, cubre todas las opciones posibles que el usuario puede interactuar, explicando los parámetros utilizados en el desarrollo SOWMEIM v1.0 como también para las diferentes opciones, mensajes, ventanas emergentes y botones. Incluye comentarios teóricos y básicas técnicas de programación para aclarar algunos puntos de interés.

FUNCIONES DE UTILIDAD

MATLAB permite desarrollar programas con el aspecto típico de las aplicaciones de Windows. Para todos los controles, se utilizará la función **uicontrol**, que permite desarrollar dichos controles.

Existen ocho tipos de controles diferentes: pushbuttons, checkbox, radio buttons, sliders, pop-up menus, editable textboxes, static textboxes y frames. La utilización de cada uno de ellos vendrá en función de sus características y aplicaciones.

DEFINICIONES BÁSICAS

Brillo: Indica si un área está más o menos iluminada.

Color: El color se forma mediante la combinación de los tres colores básicos rojo, azul, y verde (en inglés RGB).

Contraste: Es la inexistencia o escasez de tonos intermedios, de tal manera que resaltan mucho lo claro y lo oscuro, es decir, el contraste es el que permite diferenciar visualmente objetos plasmados en una imagen; está relacionado con la

diferencia de los niveles de gris claros y oscuros, la modificación del contraste consiste en variar el rango dinámico de la imagen

Histograma de una imagen: Es una representación del número de píxeles de cierto nivel de gris en función de los niveles de gris.

Píxel: Elemento básico de una imagen (Picture element).

Ruido: Es la manifestación de píxeles aislados que toman un nivel de gris diferente al de sus vecinos.

Todas las imágenes tienen cierta cantidad de ruido, la cual se puede deber a la cámara o al medio de transmisión de la señal. El ruido puede clasificarse en los siguientes tipos:

- **Gaussiano:** Produce pequeñas variaciones en la imagen; generalmente se debe a diferentes ganancias en la cámara, ruido en los digitalizadores, perturbaciones en la transmisión, etc.
- **Impulsional (sal y pimienta):** El valor que toma el píxel no tiene relación con el valor ideal, sino con el valor del ruido que toma valores muy altos o bajos (puntos blancos y/o negros) causados por una saturación del sensor o por un valor mínimo captado, si se ha perdido la señal en ese punto.
- **Multiplicativo:** La imagen obtenida es el resultado de la multiplicación de dos señales.

VENTANA PRINCIPAL (Main windows)

La Figura 23.0.- es una impresión de pantalla de la ventana principal del software SOWMEIM v1.0. Debido a la utilización del software en alumnos de la especialización en Diseño Gráfico de nuestra universidad, se decidió la terminología de la herramienta en lengua castellana, por ser un lenguaje nativo y que domina en la mayoría de los estudiantes.

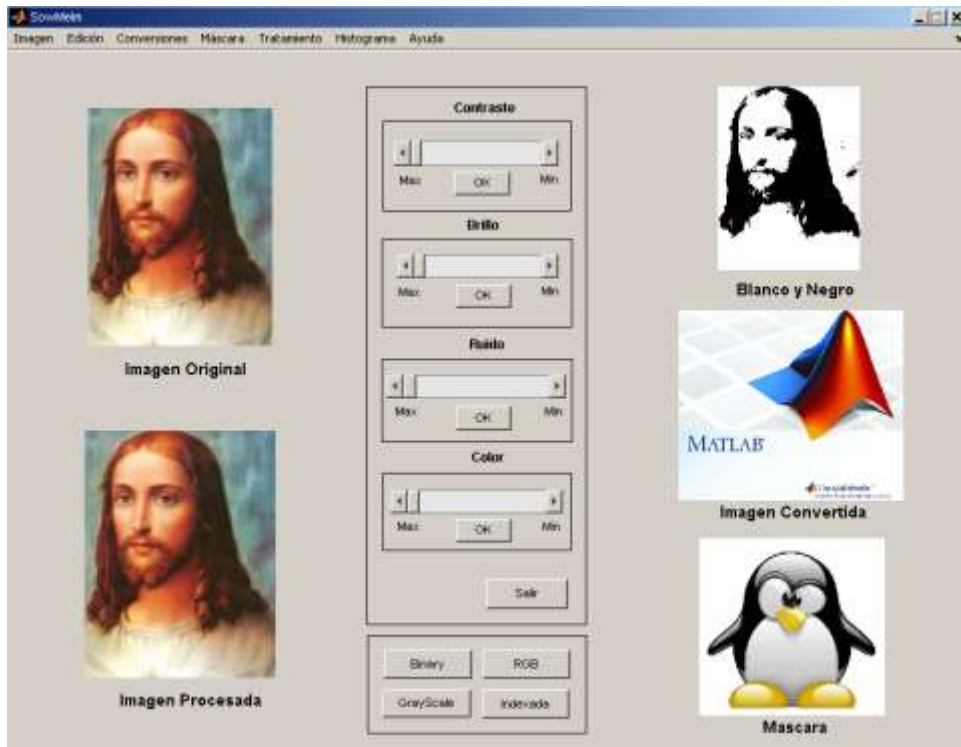


Figura 23.0: Interfaz gráfica (SOWMEIM v1.0).

MENÚ PRINCIPAL

La barra de herramientas de la ventana principal, se desglosa en 7 menús, cada uno de ellos con sus respectivos submenús; estas son las opciones:

MENÚ IMAGEN:



Figura 24.0: Rutinas del menú Imagen.

a).-Ítem Capturar: Al pulsar sobre el ítem capturar se mostrará una ventana indicando la imagen a capturar por la cámara digital; misma que me permitirá ser trasladada al interfaz del software en el campo de imagen original, gracias a la opción mostrar del sub-menú capturar.

b).-Ítem Abrir: Al pulsar sobre este ítem se abrirá una ventana de diálogo con los directorios correspondientes a las carpetas donde se encuentran imágenes con las extensiones admisibles en el software, es decir, existe la posibilidad de trabajar directamente con imágenes ya guardadas en el disco.

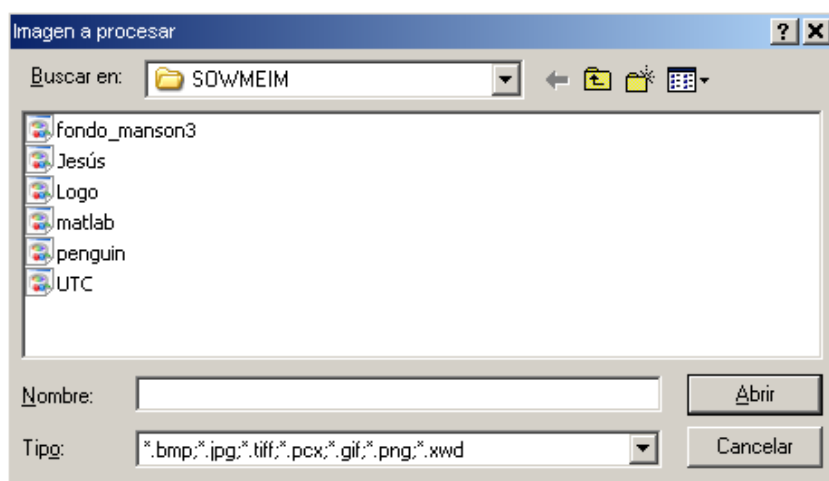


Figura 24.1: Ventana del ítem Abrir.

c).-Ítem Guardar: Con este ítem nosotros tenemos la facilidad de guardar nuestra imagen procesada y mejorada, ya que se abrirá una ventana de diálogo con el directorio correspondiente a la carpeta del software, creada en el momento de la instalación.

d).-Ítem Salir: Si el usuario quiere salir del software SOWMEIM v1.0 debe pulsar sobre este ítem y aparecerá en pantalla un diálogo en el que se les preguntará a modo de reconfirmación: 'si están seguros en salir de la aplicación'. Como se muestra a continuación:

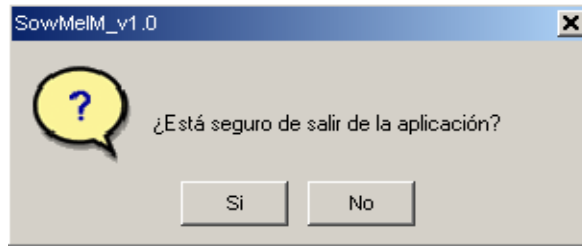


Figura 24.2: Ítem Salir.

MENÚ EDICIÓN:

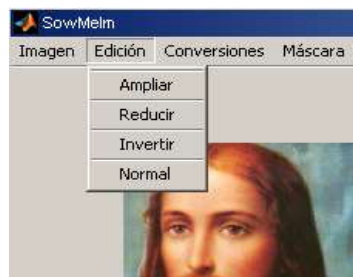


Figura 25.0: Rutinas del menú Edición.

Para realizar una ampliación, y reducción es importante conocer que interpolación es el proceso en el cual se estiman los valores de una imagen en una sección específica, cuando por ejemplo, se cambia el tamaño de una imagen y en la nueva imagen existen más píxeles que en la imagen original.

Dentro de MATLAB los comandos *imresize* e *imrotate* utilizan interpolación bidimensional como paso intermedio en sus procesos. Existen tres métodos de interpolación:

- **Vecino más próximo:** al píxel interpolado se le asigna el valor del píxel que corresponde.
- **Interpolación bilineal:** el valor del píxel interpolado es el promedio ponderado de los píxeles de la vecindad 2x2 más cercana.
- **Interpolación bicúbica:** el valor del píxel interpolado es el promedio ponderado de los píxeles presentes en la vecindad 4x4 más cercana.

Para el cambio de tamaño de una imagen, se utiliza el comando *imresize*. Este comando permite especificar: el tamaño de la imagen de salida, el método de interpolación utilizado y el filtro a usar para evitar el efecto alias. El efecto alias se presenta al reducir el tamaño de una imagen. Esto es debido a que se presenta una pérdida de información cuando se reduce el tamaño de una imagen.

a).-Ítem Ampliar: Al escoger el ítem ampliar nuestra imagen se agrandará a una determinada escala previamente ya establecida en el algoritmo, mismo que será al tamaño de un fondo de escritorio.

b).-Ítem Reducir: De la misma manera que el ítem anterior al elegir este proceso nuestra imagen se contraerá aun tamaño previamente enmarcado dentro de la programación y es al tamaño más mínimo posible tipo wallpapers.

c).-Ítem Invertir: Con este ítem el usuario puede invertir su imagen, si así lo desea.

d).-Ítem Normal: Si el usuario quiere que su imagen vuelva al tamaño normal este es el ítem que hace posible eso.

MENÚ CONVERSIONES:

Los algoritmos permiten seleccionar para la imagen original, abierta previamente, el tipo de imagen con la que se desea trabajar, haciendo una conversión de la imagen original al tipo de imagen seleccionada por el usuario.

En las conversiones a los diferentes tipos datos o imagen hemos utilizado herramientas pertenecientes al Toolbox de procesamiento de Imagen de MATLAB.

Los formatos para las diferentes funciones son los siguientes:

GrayScale: La función **rgb2gray** convierte imágenes RGB o colormap en escala de grises.

$$I = \text{RGB2GRAY}(\text{RGB})$$

El código anterior convierte imágenes de color verdadero RGB a imágenes en escala de grises y el resultado lo guarda en la variable I.

Binary: La función **im2bw** convierte imágenes RGB, en escala de grises o indexadas a imágenes binarias tomando un valor de umbral.

$$BW = \text{IM2BW}(I, \text{LEVEL})$$

La imagen es convertida a blanco y negro, utilizando un valor de umbral automático. BW es la imagen resultado, I es la imagen de entrada LEVEL el valor de umbral.

Para la función de conversión en niveles de gris, binaria, RGB, indexada además del ítem respectivo podemos hacer la llamada desde el botón 'GrayScale', 'Binary', 'RGB' e 'indexada' como podemos apreciar en la Figura 26.1.



Figura 26.0: Rutinas del menú Conversiones.



Figura 26.1: Botones de conversiones de imagen.

a).-Ítem Escala de grises: Es el primer ítem del menú conversiones y es de gran importancia ya que es el que me permitirá observar el histograma de alguna imagen. Al dar un clic en esta opción mi imagen en color se convertirá en una de percepción de colores grises.

b).-Ítem Binaria: En esta categoría del menú conversiones la imagen podrá convertirse en una con percepción de solo dos colores, es decir, blanco y negro, o imagen lógica de 0 y 1.

c).-Ítem RGB: El ítem RGB permite convertir la imagen en una de color verdadero, claro siempre y cuando la imagen a convertir a este tipo de visualización, sea una distinta a una de color.

d).-Ítem Indexada: El último ítem de el menú conversiones es la conversión indexada que como es lógico entender, convierte la imagen en una de tipo indexada que es de color al igual que la de tipo RGB, pero de diferente concepción lógica. Como mostraremos al final del manual.

MENÚ MÁSCARA:



Figura 27.0: Rutinas del menú Máscara.

a).-Ítem Crear: Al proporcionar un clic sobre el ítem crear, se mostrará en la opción máscara de la interfaz del software SOWMEIM v1.0, una imagen secundaria a la imagen que estemos procesando denominada máscara, misma que

tiene diferentes condiciones y parámetros planteados en el algoritmo programado para generar esta imagen.

b).-Ítem Borrar: Lo opuesto del ítem anterior realiza esta opción, ya que me ayuda a deshacer la máscara posiblemente creada con el proceso crear.

MENÚ TRATAMIENTO:

Este es uno de los menús más elaborados, de hecho es la finalidad del proyecto, ya que todo circula entorno al tratamiento y a sus posibles ediciones y modificaciones.



Figura 28.0: Rutinas del menú Tratamiento.

a).-Ítem Transformada de Fourier: Es una de las técnicas que permitirá mejorar la apariencia de la imagen original. Existen diversas operaciones matemáticas y transformadas que ayudan a mejorar la percepción de una imagen pero la más estandarizada y básica para usar es esta, y como podemos notar al usarla en el software SOWMEIM v1.0 presenta grandes condiciones al procesar la imagen.

La transformada de Fourier se encuentra implementada en MATLAB en el comando `fft` (para el caso unidimensional), `fft2` (para el caso bidimensional) y `fftn` (para el caso N-dimensional). Las transformadas inversas se encuentran en los comandos `ifft` (para el caso unidimensional), `ifft2` (para el caso bidimensional) e `ifftn` (para el caso N-dimensional).

b).-Ítem Filtros: La variedad de filtros que presenta esta herramienta son los filtros pasa bajas, pasa altas y el filtro pasa bandas, que a nuestro criterio dan deseados resultados en la depuración de errores de imágenes ruidosas y distorsionadas.



Figura 28.1: Rutinas del menú Filtros.

b.1).-Ítem Pasa bajas: Al aplicar un Filtro Pasa Bajas (FPB) a una imagen lo que tenemos como resultado es una imagen suavizada debido a la eliminación de las frecuencias altas que hay en la imagen, es decir, disminuye los cambios en intensidad entre píxeles consecutivos.

b.2).-Ítem Pasa altas: Los Filtros Pasa Altas (FPA), en contraste con el anterior, incrementa los cambios entre píxeles consecutivos, lo que se refleja en el realce de de bordes en la imagen.

b.3).-Ítem Pasa bandas: Este tipo de filtros solo seleccionan una banda de frecuencia, por lo tanto, si se define una banda muy ancha reobtendrá la mayor parte de la información contenida en la imagen. Si por el contrario, se define una banda muy pequeña, el resultado dependerá de las frecuencias que se dejen pasar.

c).-Ítem Extracción de rasgos: En este proceso, al dar un clic en el ítem el usuario podrá extraer los rasgos de la imagen, por medio de una ventana incorporada en el software SOWMEIM v1.0.

d).-Ítem Extracción de bordes: A diferencia del anterior proceso, este ítem me permite extraer los bordes de la imagen dejando a vista del usuario una imagen llena de trazos gráficos, a manera de un dibujo sin colores y con fondo negro. Para ello podemos aplicar los algoritmos sobel y canny.

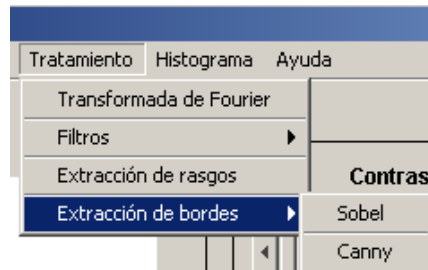


Figura 28.2: Rutinas del menú Extracción de bordes.

MENÚ HISTOGRAMA:



Figura 29.0: Rutinas del menú Histograma.

a).-Ítem Hist. Imagen Original: Esta propiedad del software prototipo SOWMEIM v1.0, proporciona en el usuario la ventaja de obtener el histograma de la imagen original simplemente con dar un clic en mencionado ítem.

b).-Ítem Hist. Imagen Procesada: Al pulsar sobre este ítem se abrirá una ventana en la cual observaremos el histograma de la imagen procesada. En dicho histograma verificaremos los niveles de grises una vez tratada la imagen.

MENÚ AYUDA:

El último menú a comentar es el Ayuda, es una nueva ventana con información sobre el contenido del proyecto en rasgos generales. Una vez dentro de la ventana siempre puedes volver al menú principal a través del botón SOWMEIM v1.0.

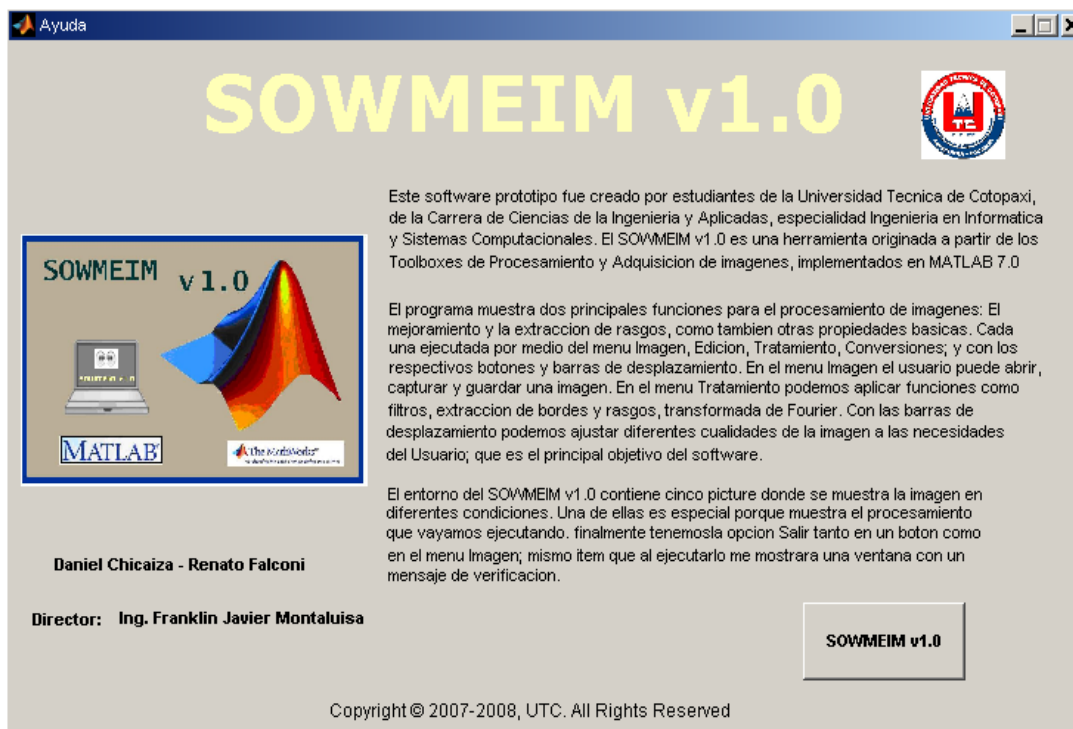


Figura 30.0: Ventana del menú Ayuda.

CONTROLES DE LA IMAGEN

Una de las propiedades del SOWMEIM v1.0 es que su interfaz consta con un frame de controles deslizadores 'sliders' que me permiten mejorar la imagen a un rango máximo, intermedio, o mínimo según la apreciación del usuario. Deslizadores incorporados para el contraste, brillo, ruido y color.



Figura 31.0: Deslizadores de control.



Figura 32.0: Ventana Extracción de Rasgos.

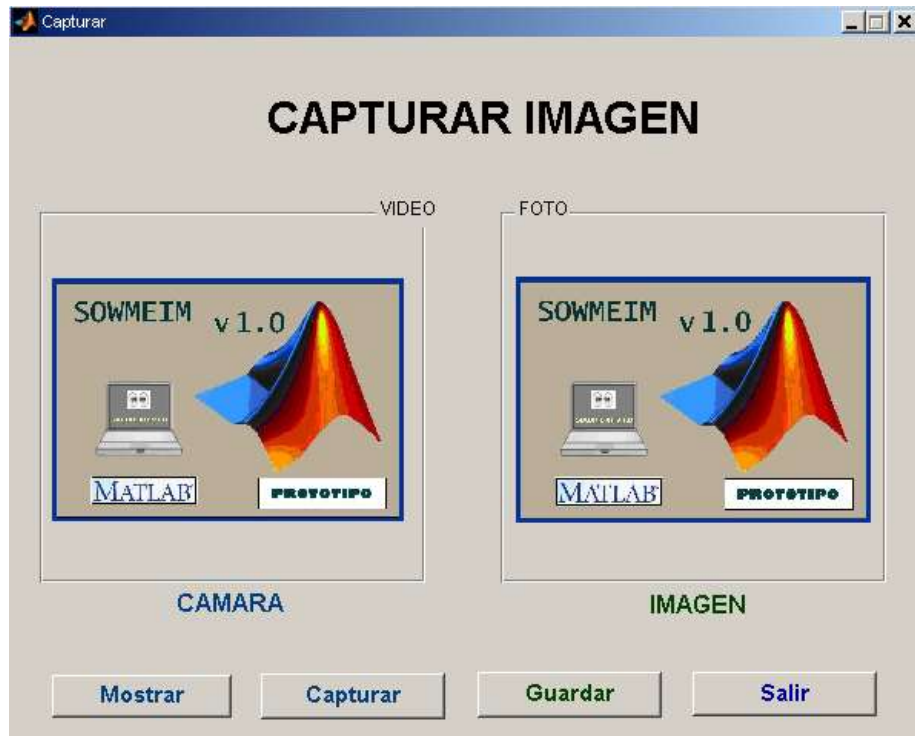


Figura 33.0: Ventana Capturar Imagen.



Figura 34.0: Ventana Aplicaciones del Procesamiento de Imágenes.

3.8 GUI con MATLAB

Según Diego Guerrero en el material “Apuntes de MATLAB” describe que el trabajo de una aplicación GUI con MATLAB, es permitir a los programadores el uso de los algoritmos implementados o diseñados por él para algún proyecto a través de una interfaz en lenguaje MATLAB, en nuestro caso, acerca del mejoramiento de imágenes digitales.

Se ha desarrollado una interfaz de acuerdo al objetivo principal de este proyecto, esta interfaz permite seleccionar la función o propiedad a ejecutar y simplemente con dar un clic en un menú o un botón la imagen captada por una cámara o extraída del disco duro, se procesaran estas automáticamente según el algoritmo seleccionado.

En este capítulo se han añadido algunas notas acerca de cómo crear todo tipo de aplicaciones GUI con la herramienta GUIDE de la versión MATLAB 7. Se espera que esto sea de gran ayuda a futuros proyectantes de la Universidad Técnica de Cotopaxi, resaltadamente para estudiantes de Ingeniería en Informática y Sistemas Computacionales que deban iniciarse en el desarrollo de este tipo de herramientas.

3.8.1 Cómo crear aplicaciones GUI con GUIDE

GUIDE (GUI Development Environment).- Es una herramienta de MATLAB para el desarrollo de aplicaciones GUI bajo un entorno gráfico. GUIDE provee al usuario de un conjunto de herramientas de uso sencillo que simplifica de enorme manera el diseño y programación de GUIs.

En primer lugar cuando el usuario abra GUIDE (tecleando por ejemplo guide desde la línea de comandos), encontrará con una ventana como la que se ve en la siguiente figura que servirá como editor de sus propias ventanas. Desde ese

preciso momento el usuario puede empezar a agregar objetos como botones, ejes y menús al área de diseño (layout area).

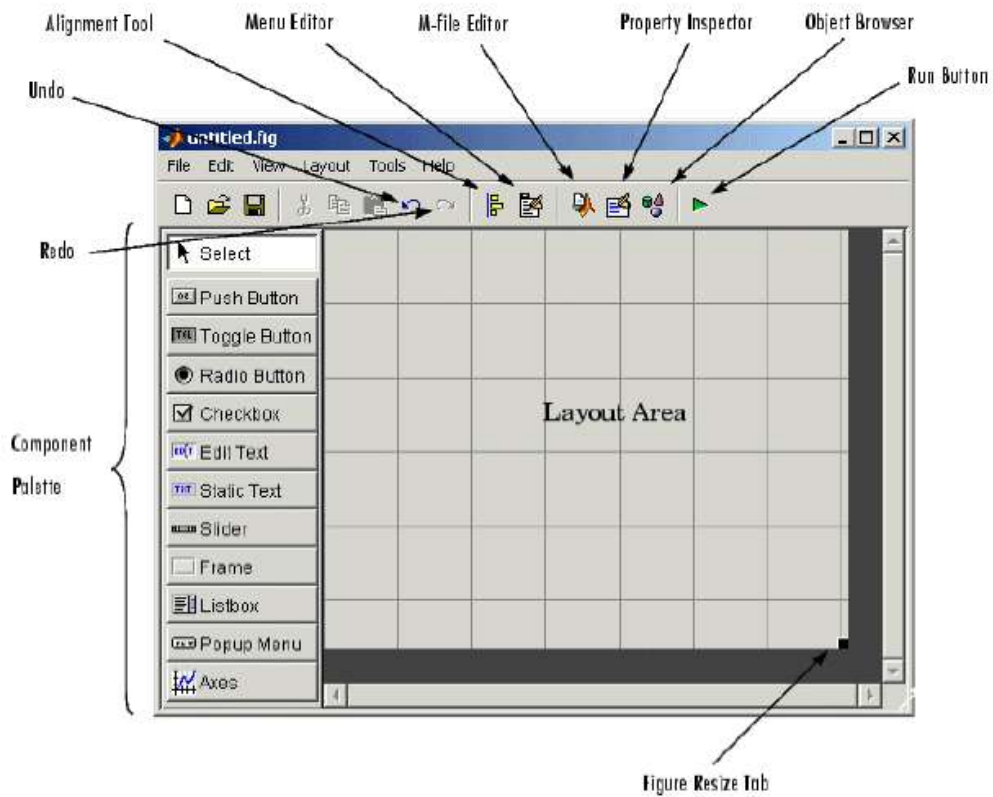


Figura 35.0: GUIDE

En MATLAB los GUI se implementan con figuras ventana que contienen lo que se llama objetos *uicontrol*, como pueden ser botones, ejes de gráfico, menús desplegables etc. Cuando se implementa una aplicación GUI se pretende programar cada uno de estos objetos según lo que se desee que ocurra cada vez que se activa ese objeto (por ejemplo cuando se pulsa un botón y se desea cerrar una ventana). Además hay que considerar que el usuario debe ser capaz de guardar y ejecutar la aplicación. Todas estas tareas se simplifican enormemente utilizando el editor GUIDE.

El desarrollo de una aplicación GUI implica básicamente dos tareas:

- Diseño de la ventana: Es decir colocar cada uno de los objetos en la situación y forma deseada.
- Programación de los componentes: O lo que es lo mismo asociar las señales emitidas por cada uno con las funciones que se desean ejecutar (*callbacks*).

Inicialmente GUIDE eran un conjunto de herramientas que permitían únicamente el diseño de la ventana, actualmente GUIDE además puede generar un archivo de extensión .M que contiene el código para controlar el lanzamiento e inicialización de la aplicación GUI. Este archivo contiene la estructura necesaria para añadir el código indispensable para la implementación de las callbacks, es decir, las funciones que se ejecutan cuando los usuarios activan los componentes del GUI.

3.8.2 Archivos generados por GUIDE

Cuando un GUI es salvado, GUIDE automáticamente genera los siguientes archivos:

- Un archivo .FIG: Este archivo contiene una completa descripción de la figura GUI diseñada, y de todos sus hijos (que pueden ser ejes o uicontrols), también contiene la información relativa a las propiedades de cada objeto. Cada vez que se retoca la figura con el editor GUIDE los cambios son guardados en este archivo.
- Un archivo .M: Este archivo contiene las funciones que controlan y ejecutan el GUI y las callbacks. A este archivo se le llama GUI M-file.

3.8.2.1 GUI M-file: La creación del GUI M-file directamente a partir de la composición de la figura simplifica enormemente el proceso de creación de las aplicaciones GUI. GUIDE genera una callback para cada componente del GUI que requiere de ellas, GUIDE solo genera la definición de esta función y debe ser el programador el que añada el código necesario en función de las necesidades de la aplicación utilizando el editor de MATLAB.

El GUI M-file además de la función callback contiene la definición de dos funciones más que se comentan a continuación.

- **Opening function:** Esta función se encarga de realizar las tareas necesarias antes de que el GUI se haga visible para el usuario, se puede utilizar esta función para realizar tareas que deben ser hechas antes del que el usuario tenga acceso al GUI, como pueden ser la lectura o escritura de datos de una fuente externa o presentar gráficos. GUIDE llama a esta función.

`mi_gui_OpeningFcn`, donde `mi_gui` es el nombre de la aplicación GUI que se está creando.

Para una aplicación GUI cuyo nombre es `mi_gui` la definición de esta función podría ser:

```
function my_gui_OpeningFcn(hObject, eventdata, handles,  
varargin) .
```

- **Output function:** Esta función saca variables a la línea de comandos si esto fuera necesario. GUIDE llama a esta función `my_gui_OutputFcn`, donde `mi_gui` es el nombre de la aplicación GUI que se está creando.

3.8.3 Estableciendo las propiedades de los componentes del GUI

Para establecer las propiedades de cada uno de los objetos que se van añadiendo a la figura de la ventana basta con señalar el objeto deseado y pinchar en la opción **Property Inspector** del menú **View**. En ese momento se mostrará una venta donde se podrán modificar las propiedades de dicho elemento.



Figura 35.1: Editor de propiedades.

Algunas de las propiedades de los componentes que cabe comentar son:

- **Name Property:** Es el de la aplicación.
- **Tags:** Sirven para identificar a cada objeto, son muy útiles a la hora de la asignación de callbacks a cada objeto, no hay que confundirlo con la cadena de texto que puede llevar cada objeto. Por ejemplo el *tag* por defecto para un botón puede ser `pushbutton1`.
- **String Property:** Es la cadena de caracteres que mostrará el componente cuando se ejecute el GUI, en el caso de un botón es el texto que mostrará y servirá para que el usuario pueda identificar su función fácilmente.
- **Pop-up Menu Items:** En el caso de que se añadan menús desplegables a la ventana el editor de propiedades sirve también para añadir los ítems o cadenas de texto que mostrará el menú.

3.8.3.1 CALLBACKS: Mediante el **Property Inspector** también se puede especificar el nombre de la callback que llevará asociada cada objeto, esta será llamada cuando el objeto sea activado.

Cuando para un objeto no se ha asociado una callback este llevará por defecto el nombre de %automática en la casilla correspondiente del editor de propiedades. En la siguiente figura puede verse esto.

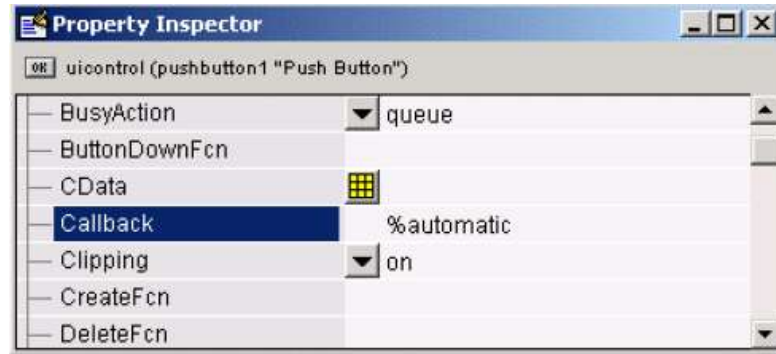


Figura 35.2: Editando Callbacks.

Cuando la figura es salvada o se ejecuta, GUIDE convierte la cadena de caracteres de la *callback* en la definición de una función que asociará al objeto en cuestión mediante el *tag* o identificador que se había especificado o si no se hizo con el que se le asigna por defecto.

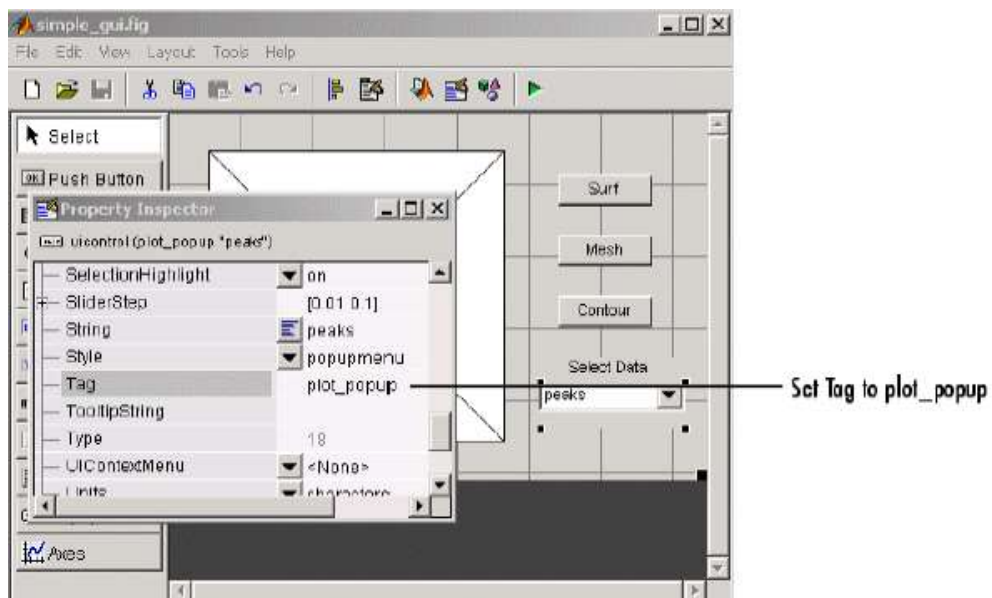


Figura 35.3: Asignación del tag de un objeto.

3.8.3.1.1 TIPOS DE CALLBACKS

1.- CALLBACK para objetos de gráficos: Todos los objetos gráficos (como pueden ser gráficas o imágenes) que se añaden al GUI tienen una serie de propiedades que permiten al programador definir funciones callbacks para asociarlas a ellas.

- `ButtonDownFcn`: MATLAB ejecuta esta callback cuando el usuario sitúa el ratón sobre el objeto y hace clic en el botón izquierdo.
- `CreateFcn`: MATLAB ejecuta la callback especificada cuando se crea el objeto.
- `DeleteFcn`: MATLAB ejecuta la callback especificada justo antes de que se borre el objeto.

2.- CALLBACK para Figuras: Las figuras (ventanas) tienen otras propiedades que pueden asociarse con sus respectivas callbacks tras las correspondientes acciones del usuario. Solamente una de estas (`CloseRequestFcn`) tiene su callback definida por defecto:

- `CloseRequestFcn`: MATLAB ejecuta la callback especificada cuando se cierra la figura.
- `KeyPressFcn`: MATLAB ejecuta la callback especificada cuando el usuario pulsa una tecla y el cursor este sobre la figura.
- `ResizeFcn`: MATLAB ejecuta la callback especificada cuando el usuario modifica el tamaño de la figura.
- `WindowButtonDownFcn`: MATLAB ejecuta la callback especificada hace clic sobre algún botón del ratón y el cursor está sobre la figura pero no sobre un objeto uicontrols activo.
- `WindowButtonMotionFcn`: MATLAB ejecuta la callback especificada cuando el usuario mueve el botón del ratón sobre la figura.
- `WindowButtonUpFcn`: MATLAB ejecuta la callback especificada cuando el usuario suelta el botón del ratón después de haberlo pulsado estando el cursor sobre la figura.

3.8.4 Administrando Datos del GUI con Estructuras de Handles

GUIDE incluye un mecanismo, llamado *handles structure*, que permite almacenar y recuperar datos compartidos utilizando la misma estructura que contiene los *handles* de los componentes del GUI. Dicha estructura, *handles structure*, que contiene los handles de todos los componentes del GUI debe ser pasada a cada callback en el archivo GUI M-file. Por lo tanto esta estructura es de gran utilidad a la hora de guardar y compartir datos. Cuando se ejecuta una aplicación GUI, el archivo .M crea la estructura de handles (*handles structure*) que contiene los datos de todos los componentes del GUI, como pueden ser menús, botones, o ejes. Cuando esta estructura es pasada como entrada a cada callback, esta se puede usar para: Compartir datos entre callbacks y Acceder a los datos del GUI. A continuación se puede ver un ejemplo de cómo se comparten datos.

3.8.4.1 Compartiendo datos entre Callbacks:

Para almacenar un dato que esta contenido en la variable X, se establece un campo de la estructura de handles y se iguala a X, después se salva la estructura con la función `guidata`.

```
handles.current_data = X;  
guidata(hObject,handles)
```

Se pueden recuperar los datos por medio de otra callback de la siguiente forma.

```
X = handles.current_data;
```

3.8.4.2 Accediendo a datos del GUI:

También se puede acceder a cualquier dato de un componente del GUI mediante la estructura de handles. Por ejemplo supóngase que el GUI tiene un componente que es un menú pop up, cuyo identificador o tag es `mi_menu`, además este

contiene tres ítems, cuyas cadenas de caracteres son abrir, cerrar y guardar. Si se quiere que otro componente del GUI como puede ser un push button ejecute un comando según la opción del menú seleccionada, en la callback asociada a ese botón habría que añadir el siguiente código.

```
opciones = get(handles.mi_menu, 'String')
opción = opciones{get(handles.my_menu, 'Value')};
```

De esta manera la variable opción toma el valor del actual valor del menú pop up, que puede ser abrir, cerrar o guardar.

Además se puede acceder a los datos del GUI entero desde la estructura de handles por medio del handle de la figura, si este fuera figura1, el código para realizar esto sería:

```
handles.figural
```

Según el grupo investigador los contenidos antes mencionados favorecen en la instauración de interfaces con la utilización de GUIDE con la herramienta MATLAB, ya que sintetiza el manejo con estructuras y variables en la programación de un GUI.

*“Visión es recuperar de la información de los sentidos
propiedades válidas del mundo exterior”*

Gibson.

“CONCLUSIONES Y RECOMENDACIONES”

CONCLUSIONES

- ☞ Con la recopilación teórica se puede deducir que una imagen para ser procesada, debe ser manipulada en forma matemática, y esto se convierte en un juego de números.
- ☞ Nuestro proceso investigativo abarca la información necesaria para que en investigaciones similares, el estudiante informático pueda reestructurarla y hacer de esta herramienta una mucho más potente.
- ☞ El mayor inconveniente que se presentó en el desarrollo sistemático de este proyecto investigativo, fue el desconocimiento de ciertas normas metodológicas indispensables en la estructura de una tesis de grado.
- ☞ Los conceptos de procesamiento de imágenes que se manejan en el mundo, son múltiples, pero pocos los planteamientos válidos para tratar una imagen.
- ☞ El proceso y manipulación con imágenes digitales en la especialización de Diseño Gráfico es la base de su formación académica.
- ☞ La elaboración de software de Inteligencia Artificial, sobre todo en el ambiente MATLAB, presenta limitaciones en su aspecto visual.
- ☞ Comprobamos que el SOWMEIM v1.0 es una herramienta que hace posible el mejoramiento de imágenes captadas por una cámara o almacenadas en el disco duro de una forma distribuida.
- ☞ Esta aplicación globaliza su uso al constar con un manual de usuario, dando apertura a una fácil utilización de aquellos que lo necesiten.

- ☞ El Ingeniero en Informática participa imperativamente en la construcción de herramientas de edición de imágenes a saber programar con matrices y vectores, sabiendo que la imagen debe ser procesada como tales.

- ☞ La construcción del software prototipo SOWMEIM v1.0 se llevo a cabo a través de una metodología de desarrollo iterativo controlado, misma que es utilizada en aplicaciones de Inteligencia Artificial.

RECOMENDACIONES

Nos permitimos recomendar los siguientes aspectos:

- ▣ El investigar los diferentes algoritmos para procesamiento de imágenes, analizarlos, revisarlos, y modificarlos serán de gran ayuda para el entendimiento de los mismos.
- ▣ El manejo eficaz de matrices y vectores en MATLAB, así como la creación de funciones promoverán el trabajo con imágenes, haciendo de esto interesante para investigar a profundidad.
- ▣ Es de gran importancia comprender las definiciones, conceptos, y argumentos de las diferentes técnicas y propiedades útiles para procesamiento de imágenes; tales como color, imagen, tipo de imagen, etc.
- ▣ Programar eficaz y pacientemente, para que al momento de depurar cierta función no haya que corregir múltiples errores.
- ▣ El SOWMEIM v1.0 en sus posteriores versiones podría ser aplicable para el control de Recursos Humanos, utilizando una Base de Datos de imágenes. Siempre y cuando se acople los módulos necesarios.
- ▣ Los Docentes de la Especialización de Diseño Gráfico pueden utilizar esta aplicación para el desarrollo de su actividad educativa.

REFERENCIAS BIBLIOGRÁFICAS

Bibliografía Citada

CERQUERA, Yamil. Curso Básico de MatLAB, Universidad Surcolombiana, Neiva – Huila. Año 2007. (Páginas 23, 35, y 49).

CUEVAS Erik, ZALDIVAR Daniel. Visión por Computador utilizando MatLAB y el Toolbox de Procesamiento Digital de Imágenes., Edición Particular., Año 2004. (Páginas 14, 15, y 22).

JÄHNE, Brend. “Digital Image Processing, Concepts, Algorithms and Scientific Applications”, 4th Edition, Springer. (Páginas 28, y 33).

Bibliografía Consultada:

CUEVAS Erik, ZALDIVAR Daniel. Visión por Computador utilizando MatLAB y el Toolbox de Procesamiento Digital de Imágenes, Edición Particular., Año 2004.

ENCISO, L. Toolbook Informático, Editorial de la Universidad Técnica Particular de Loja. Año 2006.

GUERRERO, Diego O. Apuntes de MATLAB, Universidad Técnica Particular de Loja, Grupo de Electricidad y Sistemas Electrónicos, Año 2006.

GONZÁLEZ R. C., WOODS, R. E., “Digital Image Processing”, Addison Wesley, 1993.

MERY Domingo, Visión Artificial, Tutoría del Departamento de Informática, Universidad Santiago de Chile, Año 2002.

Microsoft Encarta 2007 – 1993 – 2006 Microsoft Corporation.

MOLINA, R. Introducción al procesamiento y Análisis de Imágenes digitales, Universidad de Granada, Granada. Año 1999.

NIXON Mark, AGUADO Alberto, Feature Extraction and Image Processing, Editorial Newnes - Elsevier (2005).

SCHALKOFF, R J. Artificial intelligence: an engineering approach. McGraw Hill International Editions. Año, 1990.

The MathWorks, Inc (2007). Image Processing ToolBox, version 6, User's Guide.

The MathWorks, Inc (2007). MATLAB, versión 6, Guía del Usuario.

WINSTON, P H. Inteligencia Artificial, 3ra. Ed., Addison-Wesley Iberoamericana, USA, Año 1994.

Bibliografía Virtual:

<http://electronica.udea.edu.co/cursos/poli.htm>. (14 de noviembre del 2007).

<http://home.talkcity.com/arpeggioav/templarhawk/InteligenciaArtificial.htm>.

(21 de noviembre del 2007).

<http://iaci.unq.edu.ar/vision/archivos/apuntes.html>. (18 de enero del 2008).

[http://www.aitex.es/estudios/Vision Artificial1.pdf](http://www.aitex.es/estudios/Vision%20Artificial1.pdf). (23 de enero del 2008).

<http://www.depi.itch.edu.mx/apacheco/expo/html/ai11/vison.html>.

(04 de febrero del 2008).

<http://www.diinf.usach.cl/~dmery>. (20 de febrero del 2008).

www.mathworks.com. (10 de marzo del 2008).

www.mathworks.es. (10 de marzo del 2008).

<http://www.monografias.com/trabajos6/inus/inus.shtml>. (18 de abril del 2008).

<http://www.monografias.com/trabajos5/matlab/matlab.shtml>.

(18 de abril del 2008).

<http://www.profc.udec.cl/sram>. (02 de mayo del 2008).

www.wikipedia.org. (14 noviembre 2007 – 17 junio 2008).

ANEXOS

ANEXO N° 1

Encuesta aplicada a los Docentes y Estudiantes de Cuarto, Sexto, Séptimo, y Octavo ciclo de la especialización de Diseño Gráfico, de la Carrera de Ciencias de la Ingeniería y Aplicadas, de la Universidad Técnica de Cotopaxi.

UNIVERSIDAD TECNICA DE COTOPAXI CARRERA DE CIENCIAS DE LA INGENIERIA Y APLICADAS INGENIERIA EN SISTEMAS

Objetivo: Implementar un software prototipo de Procesamiento y Extracción de Rasgos, para mejorar la calidad de la imagen; utilizando técnicas de Inteligencia Artificial, para la especialización de Diseño Gráfico de C.I.Y.A.

ENCUESTA DIRIGIDA A LOS SEÑORES ESTUDIANTES DE CUARTO CICLO EN ADELANTE DE INGENIERIA EN DISEÑO GRAFICO COMPUTARIZADO

Instrucciones:

- Ponga una X dentro del paréntesis, de la respuesta que usted crea conveniente.
- Favor elegir una respuesta por cada pregunta.

CUESTIONARIO

1.- Para usted una imagen digital es:

- a. Aquella que, se compone de un número finito de elementos, cada uno con un lugar y valor específicos. Estos elementos son llamados *pels*, o *píxeles*.
()
- b. Una instrucción de como colorear cada píxel de un display o pantalla.
()
- c. La reproducción de la figura de un objeto por la combinación de los rayos de luz que proceden de él, a través de una pantalla.
()
- d. Ninguna de las definiciones anteriores.
()
- e. No conoce alguna definición.
()

2.- Piensa que el tratamiento de una imagen digital, dentro del diseño gráfico computarizado, tiene prioridad:

- a. Alta. ()
- b. Media ()
- c. Baja ()
- d. No tiene importancia ()

3.- ¿Qué entiende usted por procesar una imagen digital?

- a. Trabajar con una imagen digital, cambiar y mejorar su aspecto.
()
- b. Modificar las características en los datos de una imagen digital.
()
- c. Tener los datos de una imagen digital para poder trabajar con ellos, cambiando y mejorando su aspecto.
()

4.- Para usted un computador es:

- a. Una máquina electrónica que facilita el trabajo del usuario.
()
- b. Una máquina que trabaja con información, cálculos y tareas de texto.
()
- c. Una máquina para el entretenimiento y multimedia.
()

5.- En su especialización, usted ha procesado imágenes digitales:

- a. Si. ()
- b. No. ()

¿Por qué?

6.- Sus destrezas para maniobrar programas de computadora son:

- a. Excelentes. ()
- b. Buenas. ()
- c. Regulares. ()
- d. Le es muy difícil. ()

7. Utiliza un software para procesar imágenes digitales:

- a. Siempre. ()
- b. A veces. ()
- c. Nunca. ()

8. Si usted tuviera la oportunidad de utilizar un software para procesar imágenes debería ser:

- a. Fácil de usar. ()
- b. Rápido. ()
- c. Eficiente. ()
- d. Entendible y eficiente. ()

9. Si usted adquiriera un manual para el uso de un determinado software le gustaría que sea:

- a. Completo. ()
- b. Básico. ()
- c. Específico. ()

ANEXO N° 2

APLICACIONES

Separación de color RGB

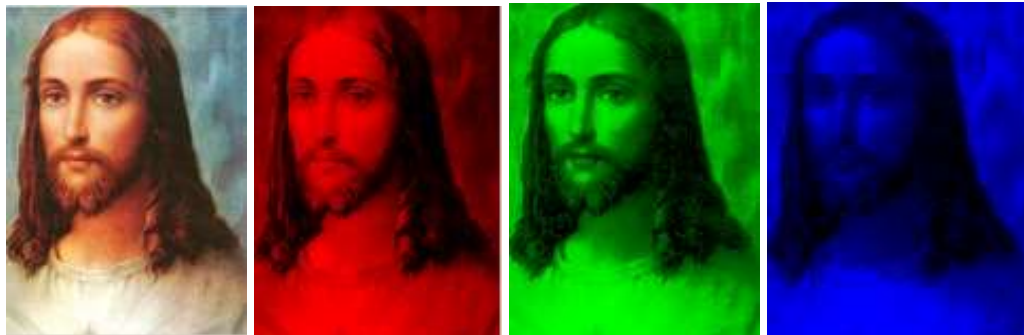


Imagen Original

Capa del Rojo

Capa del Verde

Capa del Azul

Procesados con el Cameraman



Imagen Original

Bordes disminuidos

Efecto molde

Limpieza de ruido



Imagen Original



Imagen con Ruido



Imagen limpia de ruido

Procesados con la imagen de una niña



Imagen Original



Erosión



Dilatación

Deformación, Traslado y Distorsión



Imagen Deformada



Traslado Cameraman



Distorsión Cameraman

Reconstrucción



Imagen Original



Imagen Borrosa



Imagen Reconstruida

Color RGB a color NTSC



Imagen RGB



Imagen NTSC

Suma de imágenes



Imagen Original 1



Imagen Original 2



Imagen Resultado

Cristalizado y Negativo



Imagen Original



Imagen Cristalizada



Negativo de la Imagen

Efecto publicitario



Imagen Original



Imagen preprocesado



Imagen Publicidad

Realce de imagen









Imagen Original



Imagen Mejorada

FICHA DE OBSERVACIÓN

Imágenes		Problema						
		Contraste	Brillo	Ruido	Color	Conversión	Bordes	Rasgos
1	Cameraman.tif 	Mejorado	Mejorado	-----	Mejorado	Binaria	Extraídos	Extraídos
2	Jesús.jpg 	Mejorado	Mejorado	Mejorado	-----	Escala de grises	Extraídos	Extraídos
3	Piedra.jpg 	Mejorado	Mejorado	Mejorado	Mejorado	Indexada	-----	-----

4	<p>Pout.tif</p> 	Mejorado	Mejorado	Mejorado	-----	Binaria	Extraídos	-----
5	<p>Rosanegra.jpg</p> 	-----	-----	Mejorado	Mejorado	Binaria	Extraídos	-----
6	<p>Lenna.jpg</p> 	Mejorado	Mejorado	Mejorado	Mejorado	Indexada	Extraídos	Extraídos

Elaborado por: Grupo Investigador