



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

A R A G O N

**"PERSPECTIVAS EN LA ARQUITECTURA DE
LA GRAFICACION POR COMPUTADORA"**

TESIS PROFESIONAL

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACION

P R E S E N T A N

ALEJANDRA PEREZ GARCIA

COLUMBA VARGAS MUÑOZ

ASESOR DE TESIS - ING. SILVIA VEGA MUYTOY



ENEP ARAGON

TESIS CON
FALLA DE ORIGEN

MEXICO, D.F. MARZO 1996

51
2ij

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A DIOS :

POR HABERME DADO LA VIDA, SALUD Y LO NECESARIO PARA LLEGAR HASTA ESTE MOMENTO

**A LA MEMORIA DE MI PADRE :
JUAN VARGAS ESCANILLA**

PORQUE AUNQUE YA NO ESTA CONMIGO, EN EL TIEMPO QUE ESTUVO SE ESFORZO PARA DARNIE LO MEJOR, POR HABER SIDO EL MEJOR PADRE

**A MI MADRE :
MA GUADALUPE MUÑOZ GUTIERREZ**

QUE ME HA DADO SU APOYO EN TODOS LOS ASPECTOS PARA PODER LOGRAR ESTO, Y POR NO DUDAR NINGUN MOMENTO EN QUE PODRÍA LOGRARLO POR SER LA MEJOR MADRE.

**A MIS HERMANOS :
EDUARDO, SAUL, RUBEN, ISMAEL, MARIA EUGENIA, MARIA DOLORES, IRMA Y MARIA ANGELICA**

PORQUE CADA UNO DE ELLOS DE ALGUNA FORMA ME HA MOTIVADO PARA LOGRAR ESTE OBJETIVO.

**A MIS AMIGOS :
ALEJANDRA, ANA LILIA Y JAVIER**

NO SOLO POR ESTAR CONMIGO A LO LARGO DE MIS ESTUDIOS, TAMBIEN POR SER MUCHO MAS QUE AMIGOS Y COMPARTIR MIS SENTIMIENTOS.

LILIANA, ALBERTO, MIGUEL

PORQUE ELLOS ESTUVIERON SIEMPRE CONMIGO A LO LARGO DE MIS ESTUDIOS EN LAS BUENAS Y EN LAS MALAS.

**A MI ASESORA:
ING SILVIA VEGA MUYTOY**

POR TENER LA SUFICIENTE PACIENCIA DE REVISAR ESTE TAN LABORIOSO TRABAJO Y POR SER UNA GRAN PROFESORA.

A MANUEL :

POR SU GRAN APOYO Y AYUDA DURANTE UNA ETAPA MUY IMPORTANTE DE MI VIDA.

A TODOS ELLOS LES DOY LAS GRACIAS.

COLUMBA

AGRADECIMIENTOS

A DIOS :

POR LA FE Y BENDICION QUE SIEMPRE ME HA DADO YA QUE
GRACIAS A EL HE LLEGADO A ESTE MOMENTO DE MI VIDA

A MIS PADRES :

JOEL PEREZ Y MARIA DE JESUS GARCIA
A QUIENES NO SOLO LES DEBO LA VIDA, ORIENTACION, APOYO Y CONSEJOS
CON TODO CARINO, RESPETO Y ADMIRACION

A MIS HERMANOS :

ANTONIO, OLIVA, PASTOR, TOBIAS Y MAGDALENA
POR LA CONFIANZA QUE DEPOSITARON EN MI HASTA EL TERMINO DE MI CARRERA

A MIS TIOS, PRIMOS Y MI ABUELITA :

POR EL GRAN APOYO QUE HE RECIBIDO DE ELLOS NO SOLO A LO
LARGO DE MI CARRERA, SI NO DURANTE TODA MI VIDA.

A OLIVA Y GABRIEL :

PORQUE LA DISTANCIA NUNCA HA SIDO UN FACTOR INFLUYENTE
PARA BRINDARME SU APOYO Y CARINO

A COLUMBA :

PORQUE HA SIDO SIEMPRE UNA GRAN AMIGA Y POR HABER
LLEGADO JUNTAS A ESTE MOMENTO

A MIS AMIGOS:

LILIANA, ALBERTO, MIGUEL, EUSEBIO, MATIAS
POR HABER COMPARTIDO JUNTOS TANTOS MOMENTOS DURANTE
TODA NUESTRA CARRERA, POR SU GRAN APOYO

AL ING. S. R. L. :

PORQUE SIEMPRE ME BRINDO SU APOYO PARA LA REALIZACION DE
ESTE TRABAJO. POR ESTAR EN TODO MOMENTO CONMIGO
Y RECIBIR SU CARINO.

AL M. ENC. OFELIO MARTINEZ : +
AUNQUE YA NO ESTAS CON NOSOTROS, SIEMPRE ME BRINDASTE TU
APOYO DURANTE TODA MI CARRERA, GRACIAS POR TUS CONSEJOS

A LA ING. SILVIA VEGA :
POR LA CONFIANZA DEPOSITADA PARA LA REALIZACION DE
ESTA TESIS Y POR SER UNA EXCELENTE MAESTRA.

AL ING. MANUEL MARTINEZ :
POR TODO EL APOYO RECIBIDO DURANTE TODA LA CARRERA

LES DOY GRACIAS A TODOS.....

ALEJANDRA.

OBJETIVO

El objetivo de este trabajo de tesis es conocer los conceptos básicos de lo que es la **Graficación por Computadora** y su aplicación dentro del área de Arquitectura tomando como ejemplo el diseño interno y externo de una casa habitación.

INDICE

INTRODUCCION	1
---------------------	----------

CAPITULO I

ACERCA DE LAS COMPUTADORAS	4
-----------------------------------	----------

I.1 ¿QUE ES UNA COMPUTADORA?	4
-------------------------------------	----------

I.2 HISTORIA DE LAS COMPUTADORAS	5
---	----------

I.2.1 PRIMERA GENERACION	5
--------------------------	---

I.2.2 SEGUNDA GENERACION	6
--------------------------	---

I.2.3 TERCERA GENERACION	6
--------------------------	---

I.2.4 CUARTA GENERACION	6
-------------------------	---

I.2.5 QUINTA GENERACION	7
-------------------------	---

I.3 COMPONENTES DE UNA COMPUTADORA	9
---	----------

I.3.1 HARDWARE	9
----------------	---

I.3.2 SOFTWARE	13
----------------	----

I.4 APLICACIONES DE LAS COMPUTADORAS	14
---	-----------

CAPITULO II**GRAFICACION POR COMPUTADORA 19**

II.1 DEFINICION DE GRAFICACION POR COMPUTADORA	19
II.1.1 EL DISEÑO ASISTIDO POR COMPUTADORA (CAD)	20
II.1.2 LA MANUFACTURA ASISTIDA POR COMPUTADORA (CAM)	20
II.1.3 LA GRAFICACION POR COMPUTADORA (CG)	20
II.1.4 GRAFICAS INTERACTIVAS (IG)	20
II.2 HISTORIA DE LA GRAFICACION POR COMPUTADORA	25
II.3 APLICACION DE LA GRAFICACION POR COMPUTADORA	26
II.4 PROCESAMIENTO DE IMAGENES COMO ANALISIS DE PINTURA	31
II.5 CLASIFICACION DE LOS DISPOSITIVOSGRAFICOS	32
II.5.1 DISPOSITIVOS DE ENTRADA	39
II.5.2 DISPOSITIVOS DE SALIDA IMPRESA	42
II.6 SOFTWARE UTILIZADO EN LA GRAFICACION POR COMPUTADORA	43

CAPITULO III**TECNICAS DE GRAFICACION 46**

III.1 FORMATO DE IMAGEN	47
III.2 TRAZADO DE PUNTOS	48
III.3 TRAZADO DE RECTAS	50
III.3.1 RECTAS HORIZONTALES Y VERTICALES	51
III.3.2 RECTAS DIAGONALES	52
III.3.3 RECTAS ARBITRARIAS	52

III.4 ALGORITMOS PARA TRAZOS DE LINEAS	52
III.4.1 ALGORITMO DDA	54
III.4.2 ALGORITMO DE LA LINEA DE BRESENHAM	57
III.4.3 CARGAR EL BUFFER DE ESTRUCTURA	63
III.4.4 LINEAS CON ANTISEUDONIMOS	64
III.4.5 COMANDOS DE LINEAS	67
III.4.6 LLENADO DE AREAS	69
III.5 ALGORITMOS DE GENERACION DE CIRCUNFERENCIAS	70
III.5.1 ECUACIONES DE CIRCUNFERENCIAS	70
III.5.2 ALGORITMO DE CIRCUNFERENCIA DE BRESENHAM	72
III.5.3 ELIPSES	78
III.5.4 OTRAS CURVAS	79
III.6 GENERACION DE CARACTERES	80
III.7 CONJUNTOS DE INSTRUCCIONES PARA PROCESADORES DE DESPLIEGUE	83
III.7.1 SISTEMAS DE RASTREO CON RASTREADO	83
III.7.2 SISTEMAS DE RASTREO AL AZAR	84
CAPITULO IV	
<u>TRANSFORMACIONES GEOMETRICAS BIDIMENSIONALES Y TRIDIMENSIONALES</u>	87
IV.1 DEFINICION DE GRAFICAS	87
IV.1.1 GRAFICAS BIDIMENSIONALES	88
IV.1.2 GRAFICAS TRIDIMENSIONALES	90
IV.1.3 REDUCCION DE GRAFICAS DE TERCERA A SEGUNDA DIMENSION	93
IV.2 TIPOS DE TRANSFORMACIONES	94
IV.2.1 TRASLACION	95
IV.2.2 ROTACION	96
IV.2.3 ESCALACION	97
IV.2.4 REFLEXION	98
IV.2.5 CORTE	99

INDICE

IV.3 VISTAS DE BIDIMENSIONALES Y TRIDIMENSIONALES	99
IV.3.1 VISTAS BIDIMENSIONALES	99
IV.3.2 VISTA TRIDIMENSIONAL	103
IV.4 TECNICAS DE DESPLIEGUE	104
IV.4.1 PROYECCION	104
IV.4.2 INDICACION DE LA INTENSIDAD	106
IV.4.3 SUPRESION DE LINEAS OCULTAS	107
IV.4.4 SUPRESION Y SOMBREADO DE SUPERFICIES OCULTAS	108
IV.4.5 VISTAS ESQUEMATICAS Y RECORTADAS	108

CAPITULO V

LA GRAFICACION DESDE EL PUNTO DE VISTA ARQUITECTONICO **110**

V.1 METODOS DE PRODUCCION PARA EL DISEÑO EXTERNO	111
V.1.1 DATOS DIMENSIONALES (EL MODELADO)	111
V.1.2 DATOS DIMENSIONALES (LA INTERPRETACION)	111
V.1.3 COLOCACION DE UNA SUPERFICIE MATERIAL (EL MAPEO O TRAZADO DEL DISEÑO)	112
V.1.4 COMPLETANDO LA ESTRUCTURA	112
V.1.5 ADICIONANDO LOS DETALLES	114
V.2 METODO DE PRODUCCION PARA EL DISEÑO INTERNO	122
V.2.1. DATOS DIMENSIONALES DE ENTRADA (EL MODELADO)	123
V.2.2. VERIFICACION DE LOS DATOS DE ENTRADA (LA REPRESENTACION)	123
V.2.3 ADICIONANDO TEXTURAS (EL MAPEO O TRAZADO DEL DISEÑO)	124
V.2.4 COMPLETANDO EL INTERIOR	124
V.2.5 ADICIONANDO LOS DETALLES, TERMINADO DE LA PERSPECTIVA	124

INDICE

CONCLUSIONES 131

GLOSARIO 133

BIBLIOGRAFIA 142

INTRODUCCION

El hombre viéndose en la necesidad de analizar y almacenar información en forma exacta y rápida, tuvo que idear la forma de poder lograrlo, para resolver este problema, inventó la computadora digital.

Con la aparición de ésta, gran parte del problema ya estaba resuelto, posteriormente el objetivo fue su aplicación en diversas áreas, una de ellas y en la que se volvió una herramienta de gran utilidad fue en la arquitectura, con el desarrollo de la **GRAFICACION POR COMPUTADORA**, que es uno de los campos más interesantes y que crece más rápidamente dentro de la computación. Algunos de los sistemas de computación más complejos que se usan hoy en día están diseñados para la generación de despliegues gráficos. Se conoce el valor de una figura como un medio eficaz de comunicación y como la **capacidad de conversar en forma gráfica con una computadora**.

En el presente trabajo se presentan los principios básicos del diseño, uso y entendimiento de los sistemas de gráficas, suponiendo de antemano que las personas que realicen diseños nunca antes han tenido contacto con las gráficas por computadora, pero que está familiarizado con los conceptos fundamentales de la computación, así como sus métodos. Los componentes de hardware y software de los sistemas de gráficas se examinan destacando los métodos de diseño de paquetes gráficos.

En el **Capítulo I**, se hace un estudio general de las computadoras que ilustra desde su historia, pasando por sus generaciones hasta la diversidad de áreas de aplicación.

El **Capítulo II**, muestra la definición e historia de la **Graficación por Computadora**, además de una presentación de los componentes de hardware y software de los sistemas de gráficas y sus aplicaciones.

En el **Capítulo III**, se presentan las técnicas básicas de Graficación, lo que incluye los algoritmos fundamentales para la generación de puntos, líneas, circunferencias, elipses y otros tipos de curvas.

En **Capítulo IV**, se dan a conocer las gráficas bidimensionales y tridimensionales, además de diferentes tipos de transformaciones; tales como la translación, rotación, escalación y reflexión, incluyendo las técnicas de despliegue de gráficas.

El **Capítulo V**, muestra un diseño arquitectónico del entorno interno y externo de una casa habitación, realizado por medio de la **Graficación por Computadora**.

CAPITULO I

ACERCA DE LAS COMPUTADORAS

- ¿QUE ES UNA COMPUTADORA?
- HISTORIA DE LAS COMPUTADORAS
- COMPONENTES DE UNA COMPUTADORA
- APLICACIONES DE LAS COMPUTADORAS



ACERCA DE LAS COMPUTADORAS

Hace ya algunos años que la informática dejó de ser una actividad reservada a unos cuantos profesionales para extenderse de forma acelerada y captar el interés de grandes segmentos de la población. A este hecho tan notable ha contribuido sensiblemente la aparición en el mercado de los ordenadores unipersonales o microcomputadoras, que con una continua mejora de sus presentaciones e interconexión de sus redes de cálculo, unido a una notable reducción de precios, ha puesto al alcance del consumidor, un potente instrumento que hace algunos lustros valía cientos de miles de pesos.

En este primer capítulo introduciremos la definición de la computadora, la cual, en algunos textos la definen como un ordenador.

I.1 ¿QUE ES UNA COMPUTADORA?

Según la definición de algunos textos :

"Un equipo de cómputo, es el conjunto de dispositivos con funciones diferentes, pero relacionadas entre sí, para realizar un proceso determinado"¹.

"Es cualquier dispositivo capaz de aceptar información, aplicándole a ésta procesos establecidos y proporcionando los resultados de éstos. En forma más específica, es un dispositivo para realizar secuencias de operaciones aritméticas y lógicas; a veces, y aún más específicamente, se le llama así, a la computadora digital con almacenamiento de programas, capaz de realizar secuencias de instrucciones

¹ Arechiga R. "Fundamentos de computación". México, 1978

almacenadas internamente, en la oposición a las calculadoras en las cuales la secuencia se proporciona manualmente o bien por medio de cuentas o tarjetas"².

"Una computadora es una máquina electrónica que permite el tratamiento automático de la información. Una computadora se diseña o se configura para satisfacer las necesidades de cada usuario"³.

I.2 HISTORIA DE LAS COMPUTADORAS

La computadora es un dispositivo que con el tiempo ha ido evolucionando. Al principio eran grandes y difíciles de manejar, en la actualidad son pequeñas y fáciles de procesar que cualquier persona es capaz de manejarlas.

La evolución de las computadoras se divide en las siguientes generaciones, contando con la actual:

1. Primera Generación.
2. Segunda Generación.
3. Tercera Generación.
4. Cuarta Generación.
5. Quinta Generación.

I.2.1 PRIMERA GENERACION

La primera generación de las computadoras comprende de 1945 a 1951-53. La constituye la llamada UNIVAC I, estas se caracterizan por utilizar válvulas o tubos de vacío, no se disponía de programas de apoyo y sus equipos periféricos eran lentos y poco eficaces. Por todo esto, estas eran muy costosas, tenían muchas probabilidades de no tener un buen funcionamiento, era difícil su uso, consumían gran cantidad de energía, disipaban mucho calor y eran lentas.

Su campo de aplicación era exclusivamente científico y militar.

² Landen H. N. Gildersleeve T. R. "Diseño de sistemas de computación". Enero, 1979

³ Joyanes Aguilar Luis "Programación BASIC para microcomputadoras". Agosto, 1987

1.2.2 SEGUNDA GENERACION

La segunda generación de las computadoras comprende de 1952-54 a 1963-65 y esto vino con el descubrimiento del transistor que fue el que sustituyó a la válvula de vacío, el cual acrecentó la potencia y la velocidad de estas.

En la segunda generación de las computadoras se empezó a contar con periféricos más adecuados, apareciendo lenguajes de programación que facilitaron su uso y el costo era más accesible. Con el transistor se logró mayor confiabilidad, menos pérdida de energía en calor, mayor velocidad de proceso y menor espacio requerido. En esta generación se empezó a usar la computadora en los negocios.

1.2.3 TERCERA GENERACION

La tercera generación es constituida dentro de los años de 1962-65 en adelante, y es donde la computadora en vez de usar transistores utiliza circuitos integrados que ocupan un espacio físico más pequeño. Por otro lado, es donde se generaliza el concepto de Sistema Operativo o programas básicos del control de equipo.

También mejoran los dispositivos periféricos, de lo que se obtiene mayor rapidez, menos costos, menor gasto de energía, menores requerimientos de espacio y mayor facilidad de uso, por lo que se obtuvo un aprovechamiento más eficiente.

1.2.4 CUARTA GENERACION

La cuarta generación no tuvo un período exacto donde surgiera, e incluso existen diferentes opiniones en torno a la definición de los elementos que la distinguen.

A finales de los años setenta, generalmente se acepta la aparición de la cuarta generación. La principal característica de estas computadoras es la introducción de los circuitos integrados a muy alta escala. Con esta tecnología se logra una muy alta densidad de circuitos, de 100 000 componentes o más por chip, con esto, las velocidades de proceso de las computadoras de la cuarta generación se miden en un rango de 1 a 10 nanosegundos.

Junto con estos adelantos en circuitos integrados, se desarrollaron nuevas técnicas y medios de almacenamiento tales como minidiscos y cartuchos magnéticos, también se desarrollan los sistemas de impresión de alta calidad a gran velocidad

Todo esto, propició un mejor acceso y una facilidad de adquisición de las computadoras. La reducción del tamaño de sus elementos y su menor costo permitieron el surgimiento de equipos pequeños orientados al uso personal.

En cuanto al software, las computadoras de la cuarta generación se distinguen por el surgimiento de mejores y nuevos lenguajes de programación y eficientes paquetes de aplicación de uso inmediato.

Las microcomputadoras son el equipo que se distingue durante esta generación. Surgen a principios de la década de los ochenta, y éstas provocan que puedan ser utilizadas, en la música, el diseño gráfico, la pintura, la escritura de texto, etc.

1.2.5 QUINTA GENERACION

Los japoneses son los primeros en presentar, a principio de la década de los ochenta, un proyecto de desarrollo denominado "Quinta generación". Este proyecto conlleva el intento de producir computadoras inteligentes, sistemas que se puedan programar con lenguajes naturales mediante los cuales sea posible conversar. Las características de la quinta generación son:

- a) Nuevas tecnologías de fabricación, basadas en sustancias posiblemente diferente al silicón.
- b) Desarrollo de lenguajes simbólicos para manejar símbolos y estructuras lógicas.
- c) Énfasis en nuevas arquitecturas de máquinas enfocadas al flujo de datos.
- d) Nuevos métodos amistosos para interactuar con la computadora, como son: reconocimiento de voz y procesamiento de lenguajes naturales.
- e) Inteligencia artificial dirigida a la resolución de problemas, la representación del conocimiento, las decisiones inferenciales y los sistemas expertos.

Las tablas I-1 y I-2 presentan las características más esenciales de las 5 generaciones :

GENERACION	CAPACIDAD DE ALMACENAMIENTO PRIMARIO	CAPACIDAD DE ALMACENAMIENTO SECUNDARIO	VELOCIDAD DE ACCESO A MEMORIA	EQUIPOS PERIFERICOS
1a	2 K a 30 K	de 100 K a 20,000 K	(Orden de magnitud) 10^{-2} seg.	Lectora y perforadora de tarjetas, discos, cintas e impresoras.
2a	de 8 K a 226 K	de 800 K a 300,000 K	(Orden de magnitud) 10^{-4} seg.	Lectoras y perforadoras de tarjetas, discos, cintas, tarjetas magnéticas, cintas de papel perforado, tambores y terminales.
3a	de 16 K a 1,000 K	de 1,600 K a 10'000,000 K	(Orden de magnitud) 10^{-7} seg.	Lectoras y perforadoras de tarjetas, discos, cintas, tarjetas magnéticas, tambores, terminales remotas, terminales de acceso directo, lectores de caracteres magnéticos, pantallas de video.
4a	16k hasta decenas de miles de ks	Permanece en el rango de la 1ra a la 3a generación.	(Orden de magnitud) 10 seg.	Lectoras de cintas y discos, minidisks, minicintas, terminales remotas, multiprocesadores en red, modems especializados, impresoras de alta velocidad y diversificación de tipos, pantallas de video en color, graficadores, sintetizadores musicales, sintetizadores de voz, reconocedores de caracteres ópticos y magnéticos.

TABLA I-1

CARACTERISTICAS	PERIODO DE 1963 A 1965	PERIODO DE 1962 A 1983
Componentes Electrónicos	Microcircuitos	Microcircuitos de alta densidad
Velocidad	Microsegundos	Nanosegundos
Memoria	Núcleos magnéticos	Semiconductores
Operación	En lotes	<ul style="list-style-type: none"> • Interactivo • Proceso distribuido
Lenguajes	Compiladores	<ul style="list-style-type: none"> • Compiladores mejorados • Interpretes
Sistemas Operativos	Primitivos	Avanzados
Memorias Secundarias	Principalmente cintas magnéticas	Principalmente discos magnéticos

TABLA I-2

I.3 COMPONENTES DE UNA COMPUTADORA

A la computadora se le considera una caja negra, donde entran datos que son procesados y que además emiten información.

Los componentes básicos que forman una computadora se dividen en dos grupos :

- **HARDWARE.** Es la parte física de las computadoras, es decir son los dispositivos, los circuitos eléctricos, etc.
- **SOFTWARE.** Son los programas, los sistemas operativos, etc.

I.3.1 HARDWARE

En la parte del Hardware la computadora tiene una estructura básica como la que se muestra en la Figura I-1.

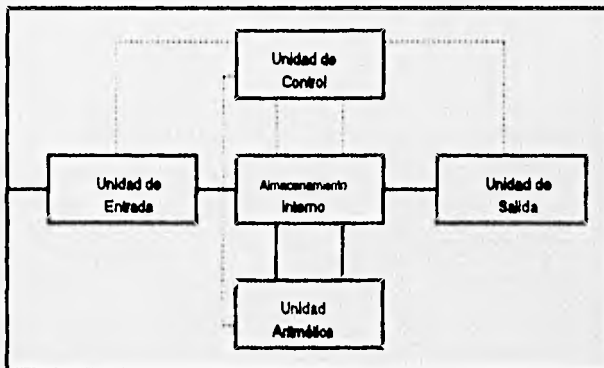


FIGURA I-1
Estructura general de una computadora

La computadora electrónica se divide en 2 partes principales :

1. El procesador central.
2. Las unidades de entrada / salida o unidades periféricas.

EL PROCESADOR CENTRAL O UNIDAD CENTRAL DE PROCESO (CPU)

La Unidad Central de Proceso es considerada el cerebro de la computadora, sirve de almacenamiento a los datos o instrucciones por procesar; permite un rápido acceso a los datos almacenados y ejerce control sobre la información; puede, además, desarrollar operaciones aritméticas, lógicas y de control; toma decisiones simples basadas en los resultados de pruebas hechas previamente; maneja la entrada de datos y la salida de la información desde los dispositivos periféricos.

El CPU tiene tres funciones principales :

- Controlar y supervisar el sistema de cómputo, con base a un programa almacenado en la unidad de memoria.
- Desarrollar las operaciones matemáticas y lógicas necesarias para procesar datos.
- Controlar el envío y la recepción de datos desde las unidades periféricas a la unidad de memoria.

Para realizar estas funciones, el CPU se vale de subunidades o subsistemas que lo componen, está compuesto de cuatro partes principales que se muestran en la Figura 1-2.

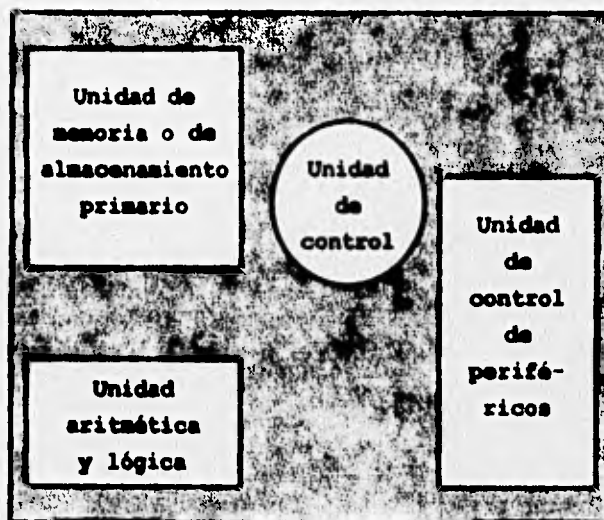


FIGURA 1-2
Diagrama Esquemático
del CPU

- a) Unidad de memoria o de almacenamiento primario
- b) Unidad de control.
- c) Unidad aritmética y lógica.
- d) Unidad de control de periféricos.

a) Unidad de Memoria o de Almacenamiento Primario

La Unidad de Memoria es la parte principal de la computadora, debido a que es en donde se encuentra la memoria y donde está contenida toda la información para la unidad de control y la unidad aritmética. En esta unidad los datos se depositan y se envían para ser procesados desde los dispositivos de almacenamiento, así también como los programas que realizarán los procesos, y los resultados obtenidos que deberán estar listos para ser enviados a un dispositivo de almacenamiento secundario.

La unidad de memoria esta compuesta por:

- a.1) Unidad de almacenamiento.
- a.2) Unidad de control y de memoria.
- a.3) Registro de dirección y de datos.

a.1) Unidad de almacenamiento.- es aquella en donde son almacenados los datos o instrucciones que son equipados en varias posiciones. Cada una de estas posiciones, tiene, dependiendo del diseño interno, un número específico de bits.

A un grupo de bits se le conoce como palabra y normalmente se transfiere y maneja con una sola operación.

El tamaño de la palabra es una característica importante en la especificación de la capacidad de la computadora debido a que determina, el tamaño máximo de la memoria que puede instalarse y la cantidad de instrucciones que puede reconocer.

Las unidades de almacenamiento son construidas con semiconductores en circuitos monolíticos de alta densidad. Las memorias semiconductoras se clasifican en:

- De solo lectura.
- De lectura y escritura.

Las memorias de sólo lectura son llamadas **ROM** (Read Only Memory) y son utilizadas para grabar los programas de uso general en forma permanente, convirtiéndose así en un híbrido entre el hardware y el software. Las segundas, son denominadas **RAM** (Random Acces Memory), estas pierden su información con la falta del suministro eléctrico, además permiten escribir, leer o modificar un dato, tantas veces como sea necesario.

a.2) Unidad de control y de memoria.- Es la que coordina en forma autónoma la unidad de memoria en su totalidad, recibe llamadas desde la unidad de control del procesador central para conectarse con otros subsistemas y poder recibir o enviar datos.

a.3) Registro de dirección y de datos.- Pueden clasificarse en tipo operacional y de almacenamiento. Un circuito operacional es capaz de almacenar información binaria y de realizar el procesamiento de la información. Un registro de almacenamiento se utiliza para almacenar de forma temporal la información, la cual no va a ser alterada ni transferida.

b) Unidad de Control

La unidad de control es en donde se supervisa la información que entra a la máquina y se decide cómo y cuándo realizar las operaciones. Le indica a la unidad aritmética qué hacer y cuando conseguir la información necesaria. Se da cuenta cuando la unidad aritmética ha terminado de realizar los cálculos y le da instrucciones de lo que tiene que hacer con respecto a los resultados y lo que deberá hacer posteriormente.

Direcciona y controla las operaciones de todas las unidades del sistema, también vigila el ciclo de la máquina.

c) Unidad Aritmética y Lógica

En ésta unidad es en donde se controlan y ejecutan las operaciones aritméticas y lógicas. Sólo ejecuta cálculos de suma, resta, multiplicación, división y operaciones relacionales (=, <, >, etc.). Esta unidad contiene:

- Circuitos.
- Registros.
- Unidades de control de procesos.
- Unidad de algoritmización..

d) Unidad de Control de Periféricos

Esta unidad controla tanto los datos que salen como los que entran a los diferentes dispositivos periféricos, así como el acceso a los mismos.

Cuando una instrucción demanda entrada o salida de datos, la unidad de control de periféricos determina si es posible enviar información desde la memoria principal hacia un dispositivo de salida o enviar datos desde un dispositivo de entrada, para ello verifica si cuenta con un canal libre. Posteriormente, determina si el dispositivo que va a ser usado está libre o no.

Algunos ejemplos de dispositivos de entrada y salida son: el teclado, las pantallas, las terminales, las impresoras, las unidades de disco, las unidades de cinta, los plotters, los lápices ópticos, los ratones, las palancas de mandos, las burbujas magnéticas, los graficadores, etc.

1.3.2 SOFTWARE

Las computadoras no podrían trabajar solo con los dispositivos mencionados anteriormente debido a que para su funcionamiento es necesario trabajar con direcciones, por lo que es indispensable introducir un componente muy importante: El Software, que como mencionamos anteriormente es la parte de la computadora que está compuesta por programas, compiladores, sistemas operativos, etc. que son los que van hacer funcionar a la computadora, por medio de una serie de instrucciones.

Los **programas** son una serie de instrucciones que realizan una función determinada. También se puede decir que es un plan para la solución de un problema.

Los **compiladores** son lenguajes de alto nivel que permiten traducir un programa escrito, a un programa que la máquina pueda entender, es decir, lo convierten a un lenguaje de máquina.

Los **sistemas operativos** son aquellos programas que le van a indicar a la computadora como debe de trabajar; es decir, es el traductor entre la computadora y la persona que la maneja. Y es quién ordena y clasifica la información.

I.4 APLICACIONES DE LAS COMPUTADORAS

Debido a que la computadora es una herramienta de cálculo capaz de procesar cualquier tipo de información y procesarla a velocidades muy altas, esta es aplicable a la solución de problemas de cualquier rama del conocimiento humano. De esta manera, actualmente se aplican en un gran número de industrias, escuelas, bancos, centros de investigación, centros administrativos, etc.

Las aplicaciones, basadas en su objetivo y en sus funciones se pueden dividir en dos.

- a) Aplicaciones Técnicas.
- b) Aplicaciones Administrativas.

aunque entre ellas siempre hay una interacción mutua.

a) Aplicaciones Técnicas

La computadora es una herramienta valiosa para estudiar las consecuencias de las leyes físicas, no sólo es capaz de evaluar expresiones matemáticas a altas velocidades sino también puede efectuar cálculos tantas veces como se desee con diferentes series de valores numéricos, de comparar los resultados y encontrar los valores óptimos que se usaron.

Las computadoras para usos técnicos, se diseñan generalmente para realizar cantidades grandes de cálculos sobre cantidades pequeñas de datos, por lo general tiene poca necesidad de contar con equipo periférico adicional complicado.

Las aplicaciones técnicas se pueden resumir en lo siguiente:

1. **Solución de problemas matemáticos.** Se han superado las barreras que representaban los métodos lentos para el desarrollo de las matemáticas con la optimización de operaciones ya concebidas y elaboradas.
2. **Programación lineal.** Es una técnica para determinar por la interacción de numerosas variables restringidas, cuáles de las líneas posibles de acción son las más factibles u óptimas.

3. **Matemáticas estadísticas.** La velocidad y la exactitud de la computadora posibilita que se realicen esfuerzos en el campo de la estadística, que sin ésta, no se emprenderían.
4. **Programación automática.** Si se le dan los medios suficientes de almacenamiento secundario, puede avanzar indefinidamente después de las instrucciones iniciales, de una operación a otra, por el simple expediente de introducir cualquier instrucción y datos que se necesitan para la debida continuación de un proceso.

Este último requisito se satisface con la creación de una biblioteca de subrutinas que pueden ser:

- Rutinas de propósitos generales (matemáticas).
- Rutinas de compilación.
- Rutinas generadoras.
- Rutinas interpretativas.
- Rutinas de edición.
- Rutinas relativas a operaciones de ingeniería.

5. **Simulación de sistemas.** Debido al avance de la técnica, se ha hecho necesario contar con herramientas de cálculo capaces de auxiliar al hombre en el diseño y simulación de sistemas.
6. **Solución a problemas de ingeniería.** Mediante la solución de ecuaciones matemáticas se ha logrado con ayuda de la computadora, resolver problemas que hasta hace pocos años era prohibitivo tratarlos, por el tiempo y los recursos necesarios para ello.
7. **Control de procesos.** Actualmente las grandes industrias cuentan con computadoras de propósitos específicos para el control de procesos, los cuales son comparados con los estándares que están dentro de la máquina. Para el caso de que éstos no coincidan, la computadora tiene un programa almacenado donde están las posibles alternativas para efectuar la corrección.

b) Aplicaciones Administrativas

La principal diferencia en el uso de la computadora en administración y en trabajos técnicos, radica en la relación de operaciones ejecutadas al total de datos procesados. Mientras que la computadora en los negocios ejecuta sólo unos cálculos para cada dato, en un problema científico, un solo dato puede emplearse en una gran cantidad de operaciones.

Uno de los primeros usos de la computadora en la administración, fue la mecanización de la mayoría de las rutinas y aspectos de oficina. Esto es en resumen, utilizar la computadora para que efectúe todo el trabajo primario.

Los campos de aplicación en el área administrativa se pueden resumir a los siguientes:

1. **Aplicación de procedimientos en uso.** La computadora se puede emplear para elaborar un trabajo que se esté efectuando por otros medios.
2. **Utilización en trabajos más allá de los trabajos presentes.** Uno de los hechos más notables acerca del uso de las computadoras en los negocios, es que abre perspectivas nuevas y más amplias en sistematización de datos en varios sentidos. En primer lugar, la gran velocidad y bajo costo de mover información eleva cualquier cifra tope anterior, relativa a la cantidad de datos factibles de manejo. En segundo lugar, puede producir mucho más información oportuna. El tercer lugar, el trabajo de la computadora aumenta generalmente la comprensión de los procesos y contribuye a la mejor orientación de los modos de organizar las funciones del negocio. En cuarto lugar, se aumenta la probabilidad de consolidar e integrar, e incluso de reducir y eliminar diversas operaciones. En quinto lugar, se puede obtener información nueva que estimule el desarrollo de nuevos planos de referencia para el incremento de la actividad administrativa dentro de ellos.

CONCLUSIONES

De lo anterior podemos decir que una computadora es una máquina electrónica que junto con otros dispositivos como lo son el teclado, el monitor, etc., es capaz de realizar por medio de una secuencia de pasos, una serie de objetivos que son fijados por el usuario. Las computadoras han pasado por una serie de generaciones; en la primera generación las computadoras eran muy

grandes, costosas y difíciles de manejar, actualmente son pequeñas, de bajo costo y muy fáciles de manejar.

Las computadoras están compuestas de dos componentes básicos; uno el **Hardware**, que son todos aquellos dispositivos electrónicos y el **Software** que son los programas, sistemas operativos, etc. Las computadoras se han vuelto una herramienta muy útil para diversas áreas, y por su objetivo y funciones se pueden dividir en dos grandes aplicaciones. Las técnicas, donde el uso de la computadora es para estudiar las leyes físicas, evaluar expresiones matemáticas y efectuar cálculos cuantas veces sean necesarios. Las administrativas, que utilizan a la computadora para mecanizar las rutinas de oficina.

CAPITULO II

GRAFICACION POR COMPUTADORA



- DEFINICION DE GRAFICACION POR COMPUTADORA
- HISTORIA DE LA GRAFICACION POR COMPUTADORA
- APLICACION DE LA GRAFICACION POR COMPUTADORA
- PROCESAMIENTO DE IMAGENES COMO ANALISIS DE PINTURA
- CLASIFICACION DE LOS DISPOSITIVOS GRAFICOS
- SOFTWARE UTILIZADO EN LA GRAFICACION POR COMPUTADORA

GRAFICACION POR COMPUTADORA

Las computadoras se han convertido en una herramienta poderosa para la producción rápida y económica de ilustraciones. Los objetos gráficos pueden ser imágenes fotográficas, o pueden ser creadas con ayuda de la computadora en forma de caracteres alfanuméricos, símbolos especiales, líneas dibujadas o áreas sombreadas de gris semejante a objetos creados artificialmente, estos pueden ser representados en blanco y negro.

Prácticamente no existe ninguna área en la cual no puedan utilizarse los despliegues gráficos con alguna ventaja. Aunque las primeras aplicaciones en ciencia e ingeniería tenían que basarse en equipos costosos y complicados, los adelantos en tecnología de computación han hecho de las Gráficas por Computadora interactivas una herramienta práctica. Se puede advertir que estas gráficas se utilizan rutinariamente en áreas diversas como la administración, la industria, el gobierno, arte, entretenimiento o esparcimiento, publicidad, educación, investigación, capacitación y medicina.

II.1 DEFINICION DE GRAFICACION POR COMPUTADORA

Desde que la **Graficación por Computadora** es relativamente una nueva tecnología; un número de términos y definiciones son usados en este campo, tales como **el Diseño Asistido por Computadora (CAD)**, **las Gráficas Interactivas (IG)**, **la Graficación por Computadora (CG)**, y **la Manufactura Asistida por Computadora (CAM)**. De estos términos CAD es el significado más general.

II.1.1 EL DISEÑO ASISTIDO POR COMPUTADORA (CAD)

CAD puede ser definido como un uso de la computadora para asistir en el diseño de una parte individual de un subsistema o de un sistema total, su uso no incluye gráficas. El proceso de diseño puede ser en el nivel de concepto del sistema o en el nivel de la parte detallada del diseño. **CAD** puede ser también una interface con **CAM**.

II.1.2 LA MANUFACTURA ASISTIDA POR COMPUTADORA (CAM)

La Manufactura Asistida por Computadora es el uso de una computadora para asistir en la manufactura o producción de una parte exclusiva del proceso de diseño, también es una interface entre el resultado de una aplicación **CAD** y la parte de programación necesaria usando lenguajes tales como **APT** (Automatic Programmed Tools) y **UNIAPT** (United's APT).

II.1.3 LA GRAFICACION POR COMPUTADORA (CG)

Graficación por Computadora es el uso de una computadora para definir, almacenar, manipular, interrogar y presentar una salida gráfica. Esta es esencialmente una operación pasiva. Las computadoras preparan y presentan información almacenada para un observador en la forma de gráfica, donde el observador no tiene directamente el control sobre la imagen presentada.

La **Graficación por Computadora** está definida como un estilo de imagen moderna tecnológicamente para computadoras de entrada y salida, incluyendo la creación, manipulación y el despliegue de imágenes con la asistencia de la computadora. La **Graficación por Computadora** representa el más reciente desarrollo mejorando la eficiencia de la comunicación entre el ser humano y las computadoras.

II.1.4 GRAFICAS INTERACTIVAS (IG)

Las Gráficas Interactivas utilizan a las computadoras para preparar y presentar material pictórico, donde el observador puede influenciar al dibujo tal como se está presentando intercalando con el dibujo en tiempo real. Para ver la importancia de la restricción del tiempo real, consideremos el problema de un dibujo complejo de tres dimensiones compuesto de 1000 líneas rotándolo

15°seg. Nosotros veremos secuencialmente que las 1000 líneas de la imagen son representadas por una matriz de 1000 x 4 de coordenadas homogéneas y que la rotación es más conveniente por la multiplicación de la matriz de 1000 x 4 que por la matriz de transformación de 4 x 4, ya que para lograr la multiplicación de matrices se requiere de 16 000 multiplicaciones, 12 000 sumas y 1000 divisiones, si esta multiplicación de matrices es realizada por software, el tiempo es significativo para el usuario.

Se pueden considerar estructuras para preparar y presentar imágenes a un observador usando un sistema de **Graficación por Computadora**, esta se encuentra dividida en las siguientes áreas:

- Representación de imágenes para ser presentadas.
- Preparación de imágenes para presentación.
- Presentación de imágenes previamente preparadas.

Aquí la palabra imagen es usada como una colección de líneas, puntos, texto, etc., donde esta imagen puede ser tan simple como una sola línea, una curva o una representación tan compleja como un automóvil.

• REPRESENTACION DE IMAGENES PARA SER PRESENTADAS

Fundamentalmente las imágenes representadas en computadoras gráficas pueden ser consideradas como una colección de líneas y material textual. Una línea puede ser representada por las coordenadas de los puntos finales de (x_1, y_1, z_1) y (x_2, y_2, z_2) o por una sola coordenada triple (x, y, z) , y el material textual por colecciones de líneas o puntos.

La representación de los textos es mucho más complejo que los casos de las líneas curvas o puntos de matrices. Desde que el usuario estuvo interesado en un patrón de reconocimiento, en el diseño de hardware gráfico y en poner caracteres fuera de lo común, casi todos los dispositivos gráficos tienen caracteres generados en la construcción de hardware o software. La representación de las líneas curvas es usualmente realizada por su aproximamiento de las mismas, es decir, por segmentos cortos de líneas derechas.

• PREPARACION DE IMAGENES PARA PRESENTACION

Todas las imágenes consisten de puntos, donde las coordenadas de estos puntos son almacenadas en un archivo previo siendo usado para presentar la imagen, este archivo es llamado base de datos. Las imágenes muy complejas requieren de bases de datos muy complejas, por lo que se

requiere de un programa muy complejo para accederlas. Esta base de datos compleja puede involucrar una estructura de anillo o una estructura de árbol y la base de datos sólo puede contener puntos, subestructuras y otros datos no gráficos. De cualquier modo, muchas aplicaciones de gráficas por computadora involucran muchas imágenes simples para que el usuario pueda inventar con facilidad una simple estructura de base de datos que pueda ser fácilmente accesada.

Los puntos son la construcción de bloques básicos de una base de datos gráfica. Hay tres métodos básicos de instrucciones tratando un punto como un ente gráfico geométrico: mueve el haz, la pluma, el cursor, o la cabeza del plotter hacia el punto, y dibuja una línea o un punto hacia ese punto. Fundamentalmente hay dos caminos para especificar la posición de un punto, las coordenadas absolutas o relativas. En las coordenadas relativas la posición de un punto está definida por el desplazamiento dado de un punto con respecto a un punto previo y la especificación de esta posición requiere de un número entero, este número entero puede ser especificado por una palabra de computadora que es equivalente a 2^{n-1} , donde 'n' es el número de bits en la palabra.

Otra forma de especificar la posición de un punto, es el uso de las coordenadas homogéneas, el uso de este tipo de coordenadas introduce a una suma compleja. Esta forma tiene una desventaja en la representación de grandes números enteros con una computadora con un límite de palabra.

En las coordenadas homogéneas un 'n' punto es representado por 'n + 1' dimensiones, los datos tridimensionales, donde la posición de un punto esta dado por el punto triple (x, y, z), es representado por 4 coordenadas (h_x, h_y, h_z, h) donde 'h' es un número arbitrario. Si cada posición de las coordenadas representada en una computadora de 16 bits fuera menos que el tamaño de la palabra, entonces 'h' sería igual a '1' y las posiciones de las coordenadas serían representadas directamente.

• PRESENTACION DE IMAGENES PREVIAMENTE PREPARADAS

Con estos comentarios acerca de las bases de datos en memoria es necesario notar que la base de datos usada para preparar la imagen de presentación casi nunca es la misma que el archivo de muestra utilizado para presentar el dibujo. La base de datos representa la imagen total mientras que el archivo de muestra representa sólo alguna porción, vista o escena de la imagen, este archivo es creado cuando se transforma la base de datos, donde la imagen contenida puede ser cambiada, rotada, trasladada o parte de esta removida, para obtener así una perspectiva antes de que se exhiba. Estas transformaciones son realizadas mediante la multiplicación de matrices.

Una matriz de 4×4 puede ser utilizada para ejecutar algunas de estas transformaciones individuales. Cuando una secuencia de transformaciones es deseada, cada transformación individual puede ser secuencialmente aplicada a los puntos para lograr el resultado deseado, sin embargo si la cantidad de puntos fuera sustancial, esta sería ineficiente y muy demorada. Una alternativa y método más deseable es el multiplicar las matrices individuales representando juntas cada transformación requerida y entonces finalmente multiplicar la matriz de puntos por el resultado de la matriz de transformación. Esta operación de matrices es llamada **Concatenación**.

Aunque en muchas aplicaciones gráficas la base de datos es mostrada completa en otras solo es mostrada una parte. Este proceso de sólo mostrar porciones de la base de datos de una imagen completa es llamado **Windowing**. Este método no es fácil, particularmente si la base de datos de la imagen completa fue transformada. La ejecución de la operación de **Windowing** en software es generalmente tiempo consumado; ya que el tiempo dinámico real de las gráficas interactivas no puede ser posible. En cambio los dispositivos gráficos sofisticados ejecutan esta función en hardware. En general existen dos tipos de **Windowing**: **Clipping** y **Scissoring**. El primer proceso va determinando líneas o porciones de líneas involucradas en la imagen que se encuentran fuera de la ventana. Esas líneas o porciones de líneas son entonces descartadas y nos son mostradas, es decir, que las líneas no son pasadas por los dispositivos de despliegue. La técnica de **Scissoring** tiene un espacio físico muy grande que es requerido cuando se está dibujando, sólo estas líneas o segmentos de línea entre la ventana especificada son hechas de manera visible; cada línea o segmento de línea marcado fuera de la ventana es un dibujo. La técnica de **Clipping** realizada en hardware es generalmente más ventajosa, que la otra. Esta técnica efectúa un área de dibujo disponible más grande que la técnica de **Scissoring**. En la técnica de **Scissoring** las líneas o segmentos de línea que para ella son dibujos no son visibles en la ventana.

En 2 dimensiones, una ventana es especificada por valores de los límites izquierdo, derecho, arriba y abajo de un rectángulo. En la técnica de **Clipping**, es fácil si los límites del rectángulo son paralelos a los ejes de coordenadas, si este no es el caso, la rotación de la ventana puede ser compensada por la rotación de la base de datos en la dirección opuesta. La técnica de **Clipping** en 2 dimensiones es mostrada en la Figura II-1. Las líneas son retenidas, borradas o parcialmente borradas, dependiendo si ellas están completamente dentro o fuera de la ventana.

En una vista tridimensional, una ventana consiste de un cuerpo de visión, como se muestra en la Figura II-2, donde el límite más cercano es 'N', el más lejano es 'F' y los lados se encuentran en 'SL', 'SR', 'ST' y 'SB'.

Como paso final en el proceso de presentación de imágenes, es necesario transformar desde las coordenadas usadas en la base de datos de la imagen (llamadas coordenadas usuarias) a coordenadas usadas por el dispositivo de despliegue (llamadas muestras de coordenadas). En particular, esto es necesario para convertir datos de coordenadas, los cuales pasan por el proceso de **Windowing** dentro de las coordenadas mostradas, tal que la imagen aparece en alguna área especificada en un dispositivo de despliegue, esta ventana es llamada **ventana activa**. Esta **ventana activa** puede ser especificada por sus bordes izquierdo, derecho, arriba y abajo, gráficos bidimensionales y tridimensionales además de estar especificando un limite cerca y lejos.

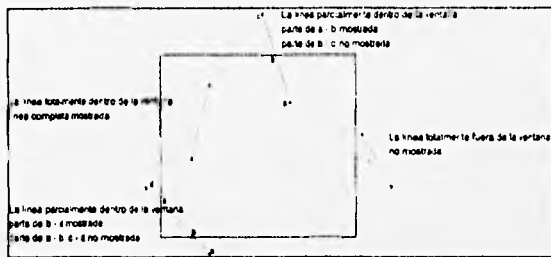


FIGURA 11-1
Bidimensional windowing
(clipping)

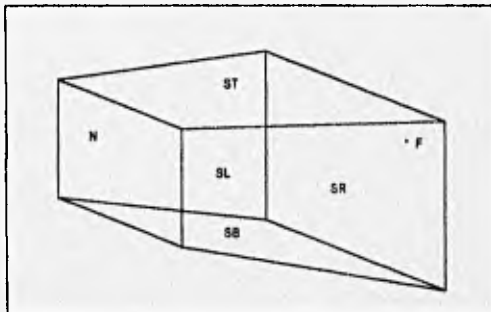


FIGURA 11-2
Cuerpo tridimensional de la visión
de la ventana

Un requisito adicional para la mayoría de las imágenes es la representación de datos de tipo carácter o alfanumérico. Existen dos métodos de generación de caracteres: El de **software** y el de **hardware**. Si los caracteres son generados por software utilizando líneas, ellos son **tratados de la misma manera** que cualquier otro elemento de imagen. En realidad esto es necesario si ellos son para ser cortados y transformados junto con los otros elementos de la imagen. Sin embargo, muchos dispositivos de gráficas tienen algún tipo de generador de caracteres en hardware. Cuando estos generadores son usados, los caracteres actuales se están generando previamente

para que se dibujen. Hasta este punto son tratados solamente como códigos de carácter. La generación de caracteres en hardware es menos flexible ya que no se permite una transformación infinita, solamente son posibles rotaciones y tamaños limitados, pero otorgan eficiencias significativas.

Cuando un generador de caracteres de hardware se está utilizando, el programa que conduce el dispositivo gráfico tiene que especificar primero el tamaño, luego la orientación y la posición donde la cadena de caracteres o texto va a comenzar, en el archivo de muestra es donde se especifican las características de los códigos de carácter. Después de que se procesa el generador de carácter, y se interpreta la cadena de texto, se busca por hardware las líneas necesarias para dibujar el carácter, y por último se dibujan los caracteres en la terminal de pantalla.

II.2 HISTORIA DE LA GRAFICACION POR COMPUTADORA

Las computadoras de los años 40's utilizaban dispositivos primitivos para la impresión, que requerían que los usuarios examinarán resultados en pilas de papel con impresiones alfanuméricas. El computador Whirlwind, construido en 1950 en el Instituto Tecnológico de Massachusetts (MIT) para la estabilidad y el control aeronáutico, fue el primer computador en incluir el tubo de rayos catódico, es decir la presentación tipo televisor.

El sistema para defensa espacial SAGE de los años 50's convertía los sonidos de detención de un radar en rudimentarias imágenes generadas por computadora. Este sistema fue el primero en emplear un lápiz óptico para seleccionar símbolos en una pantalla. Durante los años 50's, la graficación interactiva por computadora hizo pequeños progresos ya que las computadoras de ese periodo eran inadecuadas para el uso interactivo.

A fines de los años 50's y principios de los 60's, el medio de la computación en el MIT se dio a la tarea de investigar y construir un sistema computador propio. De aquí surgió en 1963 lo que vendría a ejercer la más profunda influencia en la **Graficación por Computadora** moderna: la tesis doctoral de Ivan Sutherland. El desarrolló un sistema de trazado de rectas Sketchpad, el cual permitía al usuario dibujar, por medio de una indicación en la pantalla con un lápiz óptico trazar líneas rectas y construir polígonos entre puntos, generando así diagramas complejos mediante el uso de réplicas de objetos simples.

Los primeros tubos de rayos catódicos podían trazar líneas rectas entre dos puntos en la pantalla. Pero, en virtud que una línea se desvanecía rápidamente en la pantalla, era necesario dibujarla varias veces por segundo. A principios de los años 60's, esto significaba la utilización de costosos dispositivos: memoria para almacenar los extremos de la recta y hardware para redibujarla de inmediato.

En 1968, Textronix introdujo el tubo de rayos catódico con tubo de almacenamiento, el cual retiene permanentemente un dibujo hasta que el usuario lo borra. Estos sistemas de presentación eliminaron la necesidad de recurrir a costosos sistemas de redibujado mediante el uso de memoria y hardware especiales.

La primera mitad de los años 60's marcó el inicio de un periodo de dramática reducción en los costos de ambas unidades lógicas, la memoria y el hardware. Tal reducción permitió la actual proliferación de los sistemas de presentación por barrido de memoria, en los cuales se pueden reproducir imágenes coloreadas y sombreadas con una apariencia muy realista.

En los últimos diez años, muchas personas han desarrollado importantes algoritmos usados en la graficación, con esto, el despliegue de gráficas interactivas es ahora usado en muchos países para propósitos de educación.

II.3 APLICACION DE LA GRAFICACION POR COMPUTADORA

Los diversos usos de la computadora sobre listados difieren considerablemente en una variedad de modos. Un número de criterios pueden ser usados para diferentes clases de aplicaciones. El primer criterio es el tipo de objeto e imágenes para ser producidas, este rango incluye líneas dibujadas de objetos en dos dimensiones, líneas dibujadas de objetos de tres dimensiones, las cuales son frecuentemente llamadas imágenes construidas de alambre, líneas dibujadas de objetos de tres dimensiones con eliminación de bordes ocultos como se puede observar en la Figura II-3, imágenes de tono continuo bidimensional con escala gris, imágenes de color bidimensionales y representaciones tridimensionales de objetos de sombra sólida con eliminación de superficies ocultas (Figura II-4).

Algunos de estos objetos son claramente abstractos y algunos reales, similarmemente, las imágenes pueden ser puramente simbólicas (una simple gráfica bidimensional) o muy realistas. Una segunda area de diferencia es el tipo de interacción y la serie de control sobre los objetos disponibles para el usuario

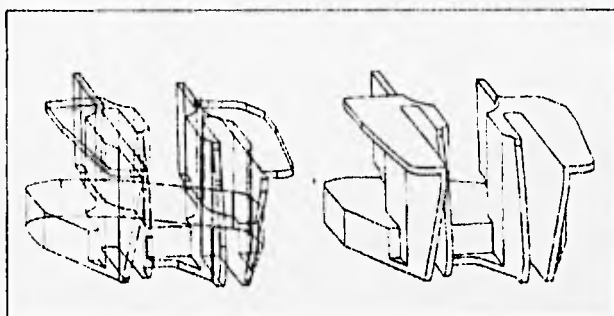


FIGURA II-3

Figura con software sólido tridimensional de un modelo geométrico. Esta secuencia es un modelo de ilustraciones de una parte mecánica, la cual fue generada por el uso del software (a) objeto dibujado en modo de estructura de alambre sin líneas ocultas; (b) una parte de la imagen con líneas ocultas



FIGURA II-4

Balero: (a) vista de un corte oculto; (b) vista de una superficie oculta

Una tercera variación es en papel de pintura o el título en el cuál la pintura es el significado hacia un fin o el fin de la misma. En cartografía, el producto primario es el dibujado, mientras que en muchas aplicaciones de diseño asistido por computadora el dibujo es simplemente una visualización de las propiedades de la geometría analizada. En esta aplicación, la fase dibujada es una importante pero pequeña parte de un largo proceso de quien el objeto es la construcción de un proceso integrado de programas de aplicación. La base de datos contiene muchas otras piezas de información orientadas a aplicaciones sobre los componentes. Estos datos son usados para

manejar otros programas de aplicación incluyendo esos para manufactura asistida por computadora

Las aplicaciones de la **Graficación por Computadora** se incrementan al ritmo de la imaginación. El diseño con ayuda de la computadora **CAD** es un término que comprende cualquier actividad en que un computador interviene en el diseño de un producto ofreciendo poderosas herramientas. El diseño de partes y el dibujo mecánico se realizan interactivamente, produciendo perfiles o producciones más realistas. Los sistemas **CAD** ayudan a los ingenieros de diseño a crear y modificar aeronaves y automóviles por ejemplo. Los dibujos elaborados con el apoyo de estos sistemas suelen pasar por varias etapas para su creación, los primeros bosquejos tienen la apariencia de figuras tipo alambre o trazado de líneas, después, durante el proceso de diseño, son detallados a fin de imprimirles realismo, de modo que el dibujo resultante puede ser coloreado o sombreado, tal como se muestra en la Figura 11-5.

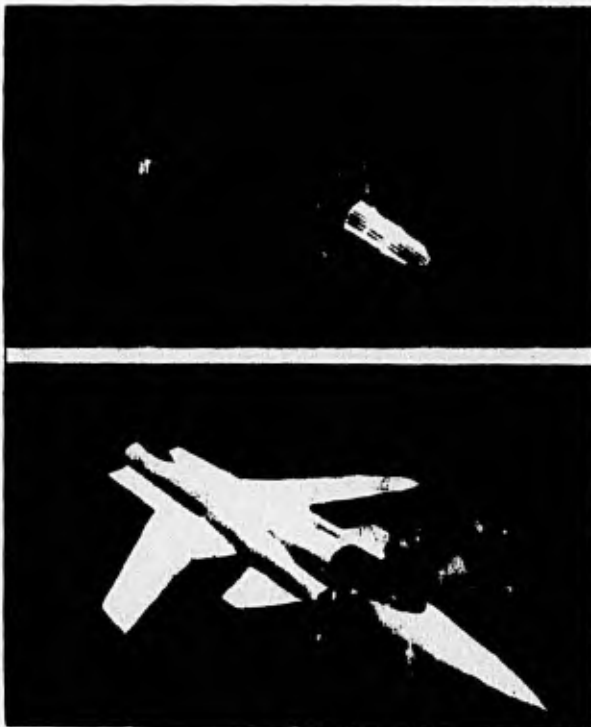


FIGURA 11-5
Interpretación de barrido de la
aeronave experimental X-29

Los sistemas CAD también han encontrado aplicaciones en el trabajo de los artistas. Cintas animadas, anuncios para televisión y aún porciones de películas cinematográficas se producen mediante estos sistemas de presentación. Por medio de técnicas particularmente sencillas usadas en secuencias de títulos de programas, el computador reproduce el contorno de los objetos. Esta animación con figuras tipo alambre es relativamente fácil de programar y requiere de una cantidad pequeña de tiempo de procesamiento. La imagen final puede ser coloreada manualmente para dar una apariencia realista. La utilización del computador para producir una secuencia animada realista exige técnicas de modelado tridimensional. Los algoritmos de interpretación deben incluir características como sombras, texturas, reflexiones y eliminación de superficies ocultas.

Los estudiantes de arte y artistas profesionales se valen también de este nuevo medio para crear dibujos realistas o abstractos. Los sistemas de diseño gráfico y de pintura, proporcionan tabletas gráficas que reproducen de manera aproximada la superficie del dibujo. La tableta es sensible a la presión de la mano, lo que produce al usuario la sensación de naturalidad al pintar. Se emula la selección de la paleta y los pinceles del pintor, permitiéndole al usuario combinar colores y diseñar pinceles con formas a su gusto. Algunos de los sistemas gráficos más complejos presentan regletas electrónicas, tijeras, pegamentos y diferentes tipos de formas. Cuando son grabadas en cintas de video o transparencias, las imágenes generadas por computador pueden pasar por las formas más tradicionales del arte.

Los pilotos de aviones y astronautas son entrenados en sistemas simuladores de vuelos gráficos generados por computadora. Esta clase de simuladores realistas son costosos, y su diseño requiere del uso de física y matemáticas muy complejas. Los simuladores de vuelo en tiempo real usan varios procesadores de propósito especial y a menudo, múltiples pantallas a fin de proporcionar al entrenado una perspectiva de 180°. Los mejores simuladores modelan escenas en aeropuertos y en diversas condiciones atmosféricas con gran precisión. El usuario de estos sistemas puede experimentar con fallas en las turbinas de despegues y aterrizajes de emergencia, colisiones en tierra o en el aire, e innumerables situaciones, sin que por ello resulten dañados la tripulación del vuelo o el simulador mismo, un ejemplo muy ilustrativo es el abastecimiento de combustible en pleno vuelo, el cual podemos apreciar en la Figura II-6.

Los procesos de manufactura también se benefician con la **Graficación por Computadora**, ya que los proyectos muestran con exactitud la forma en que se construirá el objeto. Las herramientas de la máquina numéricamente controladas se constituyen después de fabricar la parte según estos proyectos de construcción.

Los ingenieros eléctricos y electrónicos trabajan con base a los métodos de CAD más interactivos de graficas por computadora. Utilizando simbolos gráficos para representar varios componentes, un diseñador puede construir un circuito de un monitor de video, agregando componentes en forma sucesiva al proyecto del circuito.



FIGURA II-6
Programa de abastecimiento de combustible en pleno vuelo

Los diseñadores de automóviles, aviones, naves espaciales y barcos utilizan técnicas CAD en el diseño de varios tipos de vehiculos. Los trazos con estructuras de alambre (Figura II-7), se utilizan para modelar componentes individuales y planear perfiles de superficies de automóviles, aviones, naves espaciales y barcos. Las secciones de superficie y componentes de vehiculos pueden diseñarse por separado y conjuntarse para exhibir el objeto en su totalidad.



FIGURA II-7
Los sistemas CAD son usados por ingenieros para generar proyectos con estructuras de alambre de diseños de cuerpo completo

Los diseños de edificios también se crean con sistemas de gráficas de computadora. Los arquitectos diseñan interactivamente planos de pisos o plantas (Figura II-8), disposiciones de puertas y ventanas y la constitución integral de un edificio. El diseñador eléctrico puede experimentar disposiciones de cableado, tomacorrientes de electricidad y sistemas de prevención de incendios. La utilización del espacio en la oficina o en la fábrica se planea utilizando paquetes de graficación especialmente diseñados.

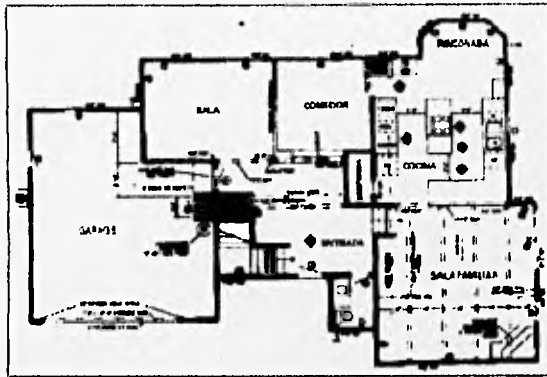


FIGURA II-8
Los sistemas CAD son usados por
arquitectos en el diseño de
proyectos de edificios

II.4 PROCESAMIENTO DE IMAGENES COMO ANALISIS DE PINTURA

La técnica de graficación que se usa para producir despliegues visuales a partir de fotografías o exploraciones de TV se llama **Procesamiento de Imágenes**. Los métodos de procesamiento de imágenes difieren de los métodos convencionales de gráficas de computadora, ya que en las gráficas de computadora tradicionales, una computadora se usa para crear la imagen, mientras que las técnicas de procesamiento de imágenes usan una computadora para digitalizar los modelos de sombreado y color a partir de una imagen ya existente. La información digitalizada se transfiere después a la pantalla de un monitor de video. Tales métodos son útiles para visualizar muchos sistemas u objetos que no se pueden apreciar directamente, como las exploraciones de TV desde una nave espacial o las imágenes visuales del ojo de un robot industrial.

Una vez que se ha digitalizado una imagen, pueden aplicarse otras técnicas de procesamiento para reacomodar las partes, realzar las separaciones de color o bien mejorar la calidad de las sombras, tal como se muestra en la Figura 11-9.



FIGURA 11-9
Fotografía borrosa de una placa de
automóvil que se vuelve legible
después de la aplicación
de métodos de procesamiento de
imágenes

El procesamiento de imágenes se utiliza ampliamente en aplicaciones de arte comercial que implican el retoque y reacomodación de secciones de fotografías y otros trabajos artísticos. En aplicaciones médicas, utilizan técnicas de procesamiento de imágenes para realizar imágenes en la tomografía. Muchos otros campos hacen uso de las técnicas de creación de imágenes para generar dibujos y analizar datos reunidos.

11.5 CLASIFICACION DE LOS DISPOSITIVOS GRAFICOS

Hay un número de métodos para clasificar los dispositivos gráficos de la computadora. Cada método otorga a veces algún arreglo confuso de dispositivos posibles, a continuación se mencionan algunos.

Primero vamos a considerar la diferencia entre un dispositivo de gráficos pasivo y uno activo. Un **dispositivo de gráficos pasivo**, simplemente dibuja imágenes bajo el control de la computadora; este permite la comunicación gráfica entre la computadora y el usuario, algunos ejemplos son el

teletipo, una impresora de línea de alta velocidad y una impresora de matriz de puntos electrostática, un plotter con una pluma y tinta y un tubo de rayo catódico con tubo de almacenamiento y refrescado (TRC). Los ejemplos de imágenes típicas que podrían generar estos dispositivos se encuentran en la Figura II-10.

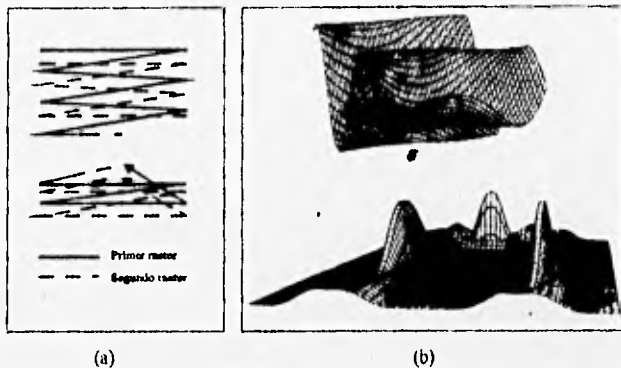


FIGURA II-10
(a) Raster, técnica de scan.
(b) típica salida de una impresora
de matriz electrostática

Un **dispositivo de gráficas activo** permite al usuario una comunicación con la computadora gráficamente. Generalmente esto implica que el usuario esté coordinando el suministro de datos de información de alguna manera indirecta. Desde una imagen curva hasta una superficie pueden ser consideradas como una matriz de datos coordinados, donde el usuario está suministrando la información pictórica. Usualmente, un dispositivo gráfico activo tiene la habilidad de repositionar el cursor y leer su nueva posición. Los dispositivos típicos de gráficas activas incluyen un cursor simple, digitalizadores o tabletas analógicas, lápices ópticos, joysticks, trackball o mouses. Aunque los digitalizadores algunas veces solamente pueden ser usados, estos dispositivos usualmente requieren de algún tipo de dispositivos gráficos pasivos para soporte. Estos dispositivos gráficos de soporte están frecuentemente basados en un TRC.

Otro método de clasificación de dispositivos gráficos son los de punto-trazado o de línea-dibujada (llamada vector). La diferencia fundamental es si está disponible un vector generador de hardware o no. Un **vector generador de hardware** permite el dibujado de líneas con una cantidad mínima de datos. Esto, por supuesto, no significa que un dispositivo de punto-trazado no pueda estar hecho para trazar vectores por software ya que un vector puede estar construido por una serie de puntos. Si los puntos son trazados de forma cerrada suficientemente juntos, el ojo los apreciará como una línea sólida. Todos los dispositivos de tubos de almacenamiento (TRC), refrescados y los plotters, son dispositivos de línea-dibujada. Algunos dispositivos

gráficos refrescados, particularmente los dispositivos raster (scan), pueden ser considerados como dispositivos gráficos de punto-trazado, así como también los teletipos, las impresoras de línea de alta velocidad, e impresoras de matriz de puntos electrostáticas. La utilización de un dispositivo puede frecuentemente estar considerado en función de su resolución; un teletipo tiene una resolución de $\pm \frac{1}{30}$ pulgadas horizontalmente y $\pm \frac{1}{12}$ pulgadas verticalmente, mientras que un dispositivo de matriz de puntos electrostático puede tener una resolución de ± 0.01 pulgadas.

Otro método para clasificar dispositivos de gráficas, requiere determinar si un dispositivo puede aceptar información tridimensional verdadera o si la información tridimensional tiene primero que estar convertida en información bidimensional. En esencia, este método requiere determinar si un dispositivo gráfico requiere tener 2 o 3 registros para obtener los datos coordinados.

En el caso de un dispositivo tridimensional, la tercera coordenada (coordenada z) es usualmente usada para controlar la intensidad del rayo TRC. Esta característica es llamada **Modulación de Intensidad o Escalación Gris** y es usada para dar la ilusión de profundidad a una imagen.

GENERACION DE UN VIDEO PARA UNA PRESENTACION

La mayoría de las pantallas de video son del mismo tipo que los tubos de rayos catódicos (TRC) (Figura II-11). El cañón de electrones contiene un cátodo que, al ser calentado, emite un haz de electrones cargado negativamente a través de una pantalla cubierta de fósforo cargado positivamente. En su trayectoria, el haz de electrones pasa por un sistema de desviación y enfoque, el cual no es otra cosa que un campo electrostático o magnético.

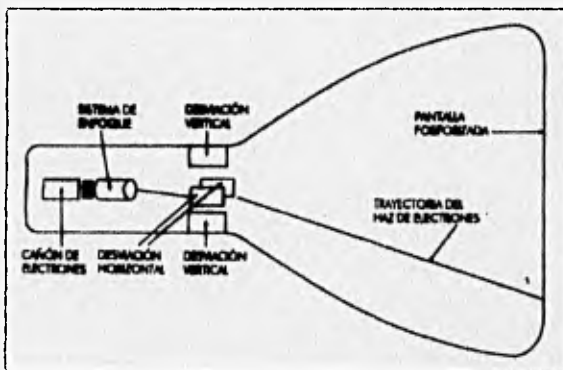


FIGURA II-11
Tubo de rayos catódicos

El sistema de enfoque concentra el haz para que en el instante que los electrones alcancen la pantalla, converjan en un pequeño punto. El sistema de desviación, que consta de dos pares de placas (horizontal y vertical), dirige el haz de electrones a cualquier punto de la pantalla. Ambos pares de placas tienen niveles de tensión iguales, pero uno tiene una carga positiva, y el otro una carga negativa.

Cuando un electrón cargado negativamente penetra en las placas, es atraído a la placa cargada positivamente, causando la desviación del electrón. El grado de desviación depende de la magnitud de la tensión de las placas horizontales y verticales.

Cuando este haz de electrones toca la pantalla, el fósforo emite un destello de luz visible cuya intensidad depende del número de electrones contenidos en el haz. Un destello blanco en la pantalla, corresponde a una ausencia de electrones enviados a esta zona. La duración de esta luz depende del tipo de fósforo que envuelve a la pantalla. A fin de dar la apariencia de una imagen sin parpadeo, cada punto iluminado en la pantalla debe ser intensificado muchas veces por segundo. Este tipo de pantallas de videos se conoce como **TRC con Mantenimiento**. Existen dos tipos de mantenimiento: **por barrido y por vectores aleatorios**.

Un TRC de color tiene tres cañones de electrones, uno para cada color primario: rojo, verde y azul. Un sistema de cañones en delta dispone los tres cañones en un patrón triangular con una rejilla perforada de metal, o máscara de sombras colocada entre los cañones y la cara de la pantalla de presentación. Cada pixel se compone a partir de un patrón triangular de puntos en rojo, verde y azul. Los orificios de la máscara de sombras se encuentran alineados para que cada cañón de electrones encienda el punto de fósforos correspondiente.

PANTALLA CON PRESENTACION DE BARRIDO

La pantalla de video se divide en pequeños rectángulos o puntos. Estos puntos se denominan como **Elementos de Imagen o Píxeles**. En realidad el TRC es una rejilla de líneas verticales y horizontales en la que cada línea horizontal se compone de píxeles. Tales líneas horizontales se conocen como líneas de barrido, en tanto que la presentación de video se denomina **presentación por barrido**. La calidad de una presentación por barrido suele describirse de acuerdo a su resolución. Cuanto mayor sea la resolución, tanto más detallada resultará la imagen. Las presentaciones de baja resolución tienen cerca de 300 líneas de barrido y cada una contiene cerca de 400 píxeles. Por su parte, las representaciones de alta resolución tienen al menos 1000 líneas de barrido, con más de 1000 píxeles por línea.

El sistema de barrido presenta la imagen en el TRC por medio de una frecuencia predefinida. Al principio, las placas de desviación horizontal y vertical presentan un píxel en la esquina superior izquierda de la pantalla. Luego la tensión de las placas horizontales cambia de manera continua, guiando el haz de electrones sobre la línea de barrido. Cuando el haz llega al final de la línea, el cañón de electrones se apaga (sin haz visible) y ambos pares de placas de desviación envían el haz al píxel de la extrema izquierda, en la segunda línea de barrido.

La tarea de iluminar píxeles apropiados requiere de una unidad de presentación que consta de tres partes: **memoria de imagen, controlador de presentación y algoritmos de conversión por barridos.**

Memoria De Imagen

Cada píxel de pantalla corresponde a una entrada particular en un arreglo bidimensional residente en la memoria, y ésta se conoce como **Memoria de Imagen o Mapa de Bits.** La tendencia actual consiste en poner la memoria de la imagen al alcance de la unidad central de procesamiento (CPU) del computador principal, con el objeto de permitir la actualización rápida de la imagen almacenada.

El término píxel se usa también para describir el renglón y la columna de colocación en el arreglo de la memoria de imagen que corresponda a la posición en la pantalla.

A cada posición de píxel en pantalla y su correspondiente localidad en memoria de imagen, corresponde a un par coordenado entero (x, y), como se muestra en la Figura II-12. El valor de 'x' se refiere a la columna, en tanto que el valor de 'y' representa la posición del renglón. Por lo general, el origen de este sistema de coordenadas aparece en la esquina inferior izquierda de la pantalla.

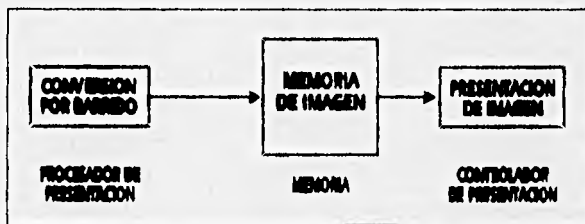


FIGURA II-12
Sistema de coordenadas de presentación

Cada pixel en el arreglo de la memoria de la imagen consta de cierto número de bits. Una imagen en blanco y negro que sólo tiene dos niveles de intensidad, encendido u apagado, tiene una memoria con un plano de un sólo bit. Para desplegar una buena imagen en color o en blanco y negro con sombra de gris, se necesitan planos adicionales de bits.

Los ocho distintos valores en tres planos de un bit son interpretados como niveles de intensidad entre '0' y $2^3 - 1 = 7$ '. Una memoria de imagen a todo color con resolución de '512 x 512' requiere $512 \times 512 \times 24 = 6'191,456$ ' bits de memoria.

Controlador De Video

El segundo componente de la unidad de presentación, es el controlador de video. Este dispositivo de hardware lee el contenido de la memoria de imagen y lo deposita en un buffer de video, para luego convertir la representación digital de una cadena de valores de pixeles en señales analógicas de tensión que se envían en serie a la pantalla de video. Siempre que el controlador encuentre el valor de '1' en la memoria de imagen, se envía una señal de alta tensión al TRC, el cual enciende el pixel correspondiente en la pantalla.

Conversión Por Barrido

La conversión por barrido es el proceso que convierte la representación abstracta de una imagen en el valor apropiado para cada pixel en la memoria de imagen. Es importante observar que cada vez que la imagen se modifica, hay que efectuar cientos de cálculos en la conversión por barrido.

PANTALLA CON PRESENTACION POR VECTORES ALEATORIOS

Si quisiéramos dibujar sólo rectas, sería preferible un sistema gráfico de trazado de rectas, que se conoce también como **sistemas de vectores aleatorios** o **sistema caligráfico**. El principio que subyace a estos sistemas es sencillo. Hay una razón de memoria, denominada **archivo de presentación**, que se compone de órdenes para el trazado de rectas.

Un procesador especial de generación de vectores por hardware interpreta esas órdenes y envía las tensiones correctas al sistema de desviación del TRC, desplazando el haz de electrones en línea recta desde el punto inicial hasta el punto final (Figura II-13). El haz se proyecta únicamente en a aquellas partes de la pantalla donde existe una línea recta o un vector.

La principal desventaja de un sistema de vectores aleatorios es su incapacidad para rellenar áreas y crear imágenes coloreadas o sombreadas realistas.

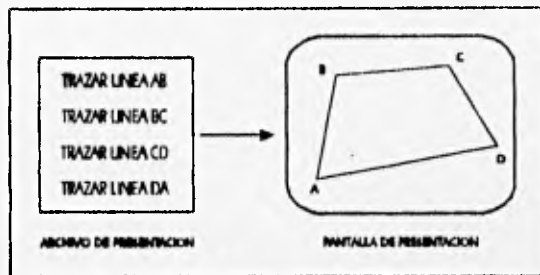


FIGURA II-13
Archivo de presentación

PRESENTACION POR TUBO DE MEMORIA

Un tubo de memoria de perspectiva directa (TMPD) es un sistema de presentación en TRC que no necesita mantenimiento desde la memoria. Detrás de una pantalla de fósforo con alta persistencia se haya una superficie de almacenamiento o rejilla (Figura II-14). Cuando el haz de electrones toca la rejilla de almacenamiento, se origina un patrón de cargas positivas. Un segundo cañón de electrones, llamado cañón de flujo continuo, emite electrones de baja energía que cubren la pantalla de manera uniforme, haciendo que la imagen almacenada por cargas positivas sea transferida a la pantalla de fósforo, dejándola visible hasta por una hora. La imagen almacenada se borra al darle a toda la rejilla de almacenamiento una carga positiva. Esto resulta en el característico destello de luz blanca que se observa antes de que la pantalla se oscurezca. Entre las ventajas de un TMPD se encuentra la alta resolución de líneas rectas dentadas y las imágenes sin parpadeo. Entre sus mayores desventajas se cuenta la falta de borrado selectivo y de actualización dinámica; cada cambio de la imagen implica borrar y redibujar toda la imagen completa.

PRESENTACION POR CRISTAL LIQUIDO

Una pantalla de representación por cristal líquido (LCD) es un tipo de tecnología de representación en tablero plano que elimina el largo cuello del TRC. Dos placas de vidrio se hallan separadas por un cristal líquido orgánico conductor. Las cargas eléctricas convierten el

liquido de un estado visible a uno invisible y a la inversa. Los sistemas de media resolución tienen una sola representación que está dividida eléctricamente en varias más pequeñas que son desplegadas simultáneamente.

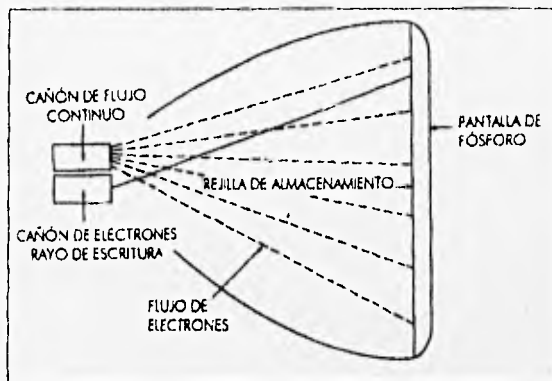


FIGURA II-14
Tubo de memoria de perspectiva directa

Las principales ventajas de las pantallas LCD es su portabilidad y su bajo requerimiento de energía. Sus desventajas son la falta de escala de grises y colores. Además, la propiedad reflectora de los cristales líquidos y los arreglos de las placas de vidrio causan un molesto reflejo del ambiente circundante.

II.5.1 DISPOSITIVOS DE ENTRADA

A continuación se listaran algunos dispositivos de entrada.

TECLADO

El dispositivo de entrada más común es el **teclado**. Se usa principalmente para introducir programas y datos en el computador. Al oprimir una tecla, un **código de carácter único** se transmite al computador. Existen 128 códigos ASCII y 256 EBCDIC, cada uno de los cuales puede usarse para emitir una respuesta diferente respecto a un programa gráfico.

PERILLAS

Una **perilla (Paddle)** es un dispositivo de posicionamiento que se encuentra conectada a un potenciómetro. Cuando la perilla gira, el potenciómetro envía una tensión a un convertidor analógico/digital, el cual transforma la tensión en un valor entero que se usa luego como valor de entrada al computador (Figura II-15). Así mismo, la mayoría de las perillas cuentan con un botón que, al ser oprimido, transmite señales que cambian un bit lógico de '0' a '1'. A menudo, se usa la perilla para desplazar un objeto en la pantalla; la dirección en que gira la perilla indica la trayectoria del movimiento. El botón, por último, se oprime cuando deseamos fijar el objeto en determinada posición de la pantalla.

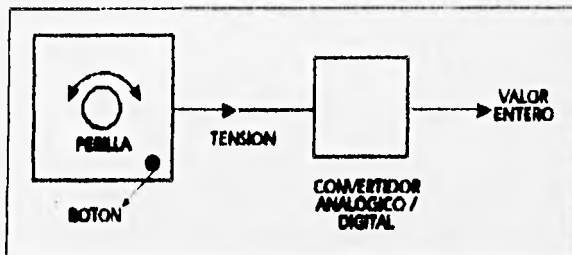


FIGURA II-15
Perilla conectada a un
potenciómetro

JOYSTICK

El **joystick** es una palanca de mando que se encuentra conectada a dos potenciómetros y que puede accionarse en cualquier dirección rotacional. Un potenciómetro registra el movimiento sobre el eje horizontal y el otro registra el movimiento del eje vertical, juntos, proporcionan valores que pueden convertirse en coordenadas de un punto en la pantalla. Una palanca de mando con botones, realiza la misma tarea que un juego con dos perillas.

LAPIZ OPTICO

Un **lápiz óptico** es un dispositivo con una fotocelda que, cuando se activa oprimiendo la punta contra la pantalla, devuelve al computador la posición del pixel iluminado en el campo visual. Lo que en realidad se envía al computador, es el tiempo de duración del ciclo de mantenimiento del controlador de video cuando el pixel es desplegado. De esta información, el computador puede

determinar la posición en pantalla del pixel, el cual debe estar iluminado a fin de poder hallarlo con el lápiz óptico. Un lápiz óptico es el que se muestra en la Figura II-16.

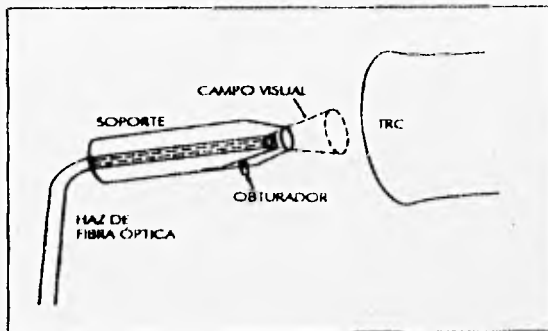


FIGURA II-16
Lápiz óptico

MOUSE

Un **mouse o ratón** es un dispositivo de aprehensión manual que aprovecha la coordinación natural entre ojos y manos del usuario, pues permite localizar y escoger de manera cómoda y precisa un objeto en la pantalla. El ratón tiene dos ruedas perpendiculares en la parte inferior conectadas a unos potenciómetros. El movimiento relativo del ratón al desplazarse sobre una superficie puede determinarse para ser usado como el valor de coordenadas en la pantalla. Al oprimir los botones que se encuentran en la parte superior del ratón se introducen valores binarios.

TABLETA GRAFICA

La **tableta gráfica o digitalizador**, es una superficie plana conectada a un cursor manual o plumilla. La plumilla se mueve sobre la superficie en la misma forma en que se dibuja sobre un cuaderno. La superficie de la tableta presenta un sistema de coordenadas similar al de la pantalla. Una posición de coordenadas se transmite al computador cuando el usuario ejerce presión de la plumilla sobre la superficie, o bien, oprime un botón del cursor. Algunas tabletas gráficas determinan puntos de coordenadas mediante una rejilla de alambres cargados eléctricamente en la superficie. Cuando la plumilla o cursor se activa, se registra la tensión en el punto de intersección más cercano de la rejilla, convirtiéndose así en coordenadas de pantalla.

II.5.2 DISPOSITIVOS DE SALIDA IMPRESA

IMPRESORA DE MATRIZ DE PUNTOS

Una **impresora de matriz de puntos**, es un dispositivo económico que produce trabajos gráficos de calidad baja y media. La cabeza de impresión móvil tiene una columna de agujas que pueden hacer presión sobre el papel por medio de pequeños electromagnéticos. Cada aguja seleccionada imprime un punto sobre el papel que corresponde a un pixel iluminado en pantalla. Una cabeza de impresión con cinco agujas se mueve a lo largo de la pantalla de graficación trazando hasta cinco renglones de pixeles de manera simultánea.

IMPRESORA DE INYECCION DE TINTA

La **impresora de chorro de tinta** (Figura II-17) emplea impulsos eléctricos para proyectar gotas de tinta sobre el papel. Cuando la cabeza de la impresora se desplaza sobre la página, lanza hasta 12 chorros de tinta en cada punto especificado. Los chorros de tinta producen imágenes sombreadas de color y de alta calidad, obteniendo así una resolución de hasta 200 puntos por pulgada.



FIGURA II-17
Impresora de inyección de tinta

GRAFICADORES

Los **graficadores** producen imágenes por medio del movimiento de una plumilla sobre el papel entre cualquier par de puntos. Si el papel no se mueve, la plumilla debe desplazarse en dos direcciones ortogonales. Si el papel se mueve, la plumilla hará lo propio en una dirección: perpendicular a la dirección del movimiento del papel. Pueden usarse plumillas de colores y espesores diferentes para producir imágenes de alta calidad.

GRAFICADORES ELECTROSTATICOS

Los **graficadores electrostáticos** son graficadores de barrido que pueden producir imágenes en color de alta calidad. Se colocan puntos electrostáticos sobre el papel cuando éste se mueve sobre una cabeza de impresión fija. A continuación el papel es expuesto a fijadores de tono líquido, los cuáles producen imágenes de color sombreadas. Si se desea usar colores, el papel debe pasar sobre la cabeza de impresión una vez para cada uno de los tres o cuatro fijadores de tono de color.

II.6 SOFTWARE UTILIZADO EN LA GRAFICACION POR COMPUTADORA

Los paquetes gráficos simples consisten de un pequeño, pero completo funcionamiento de fáciles aplicaciones independientes para la creación de vistas arbitrarias de objetos de dos dimensiones y para soportar interacciones entre el programa de aplicación y el usuario. Las subrutinas de los paquetes gráficos simples, son fácilmente desarrolladas para especificar áreas de aplicación común, datos ploteados, diseño asistido por computadora y cartografía. El software de graficación es un componente fundamental en cualquier sistema de presentación gráfica.

En 1979 el Graphics Standard Planning Committee (Comité de Planeación de un Estándar para Graficación GSPC), diseñó el Core Graphics System (Sistema Central de Graficación, CORE), con el objeto de proporcionar las capacidades necesarias para lograr el diseño de sistemas de despliegue gráfico tridimensional interactivo de alto rendimiento. Hoy en día, existen más de 200,000 usuarios del sistema CORE. Por desgracia, cada sistema difiere de los demás, lo que dificulta en el mejor de los casos, la portabilidad de los programas.

La Organización de Estándares Nacionales de Alemania Occidental, DIN, creó su propio estándar para la graficación, conocida como Graphical Kernel System (Núcleo de Sistema Gráfico, GKS). El GKS cuenta con excelentes posibilidades para la graficación bidimensional, pero carece de capacidades tridimensionales.

Los órdenes de graficación son en realidad procedimientos o funciones; un programador de aplicaciones las incluye en un programa en Pascal del mismo modo que invoca la función SIN (x) o el procedimiento READ (A, B).

CONCLUSIONES

Como se observó en este capítulo la **Graficación por Computadora**, es una nueva tecnología, la cuál está definida por diferentes términos tales como **Diseño Asistido por Computadora**, **Gráficas Interactivas**, **Graficación por Computadora** y **Manufactura Asistida por Computadora**. La **Graficación por Computadora** se inició en los años cuarenta y fue evolucionando a través de las décadas hasta el punto donde se desarrollan algoritmos usados en la graficación, por lo que la **Graficación por Computadora** tiene actualmente muchas aplicaciones importantes en diferentes áreas como la Ingeniería, la Arquitectura, la Medicina, la Publicidad, la Educación, la Aeronáutica, entre otras. Para obtener mejores resultados en la graficación, se necesita la ayuda de ciertos dispositivos de hardware y de diversos paquetes gráficos.

En el siguiente capítulo se mostrarán las técnicas de graficación utilizando algoritmos como el de Bresenham para la línea y el círculo.

CAPITULO III

TECNICAS DE GRAFICACION



- **FORMATO DE IMAGEN**
- **TRAZADO DE PUNTOS**
- **TRAZADO DE RECTAS**
- **ALGORITMOS PARA TRAZOS DE LINEAS**
- **ALGORITMOS DE GENERACION DE CIRCUNFERENCIAS**
- **GENERACION DE CARACTERES**
- **CONJUNTO DE INSTRUCCIONES PARA PROCESADORES DE DESPLIEGUE**

TECNICAS DE GRAFICACION

Las imágenes generadas por la computadora se producen usando objetos gráficos primitivos, como puntos, rectas y círculos, al igual que operaciones gráficas en pantalla, como "borrar la pantalla" o "colocar la imagen en parte de la pantalla".

La mayoría de los computadores posteriores a 1980 presentan capacidades gráficas por software y hardware que facilitan, al programador de sistemas la escritura de programas gráficos, los cuales significan el acceso al paquete de software gráfico, que no es sino una colección de subrutinas eficientes. En este capítulo se presentan algunos algoritmos de las principales rutinas primitivas de graficación, así mismo se mostrarán algunas técnicas de la graficación por medio de la computadora.

Antes de continuar, explicaremos que es una Primitiva, ya que ésta, es la base de cualquier gráfico. Se le llama Primitiva a la estructura básica de una imagen, donde se incluyen cadenas de caracteres y entidades geométricas como puntos, líneas, rectas, polígonos, circunferencias y las gráficas curvas. Las funciones de las primitivas de salida en GKS (que es un estándar para la graficación que tiene como significado Graphical Kernel System, Núcleo de Sistema Gráfico con excelentes posibilidades para la graficación bidimensional) se examinan y después se estudian los algoritmos para la generación de despliegues.

III.1 FORMATO DE IMAGEN

Si se usa el procedimiento de graficación de puntos para dibujar horizontalmente en la pantalla ocho pixeles, para después medir la recta, se encontrará que ésta tiene 0.4 cm de ancho. Si se dibujan ocho pixeles verticalmente, la línea tiene una altura de 0.3 cm (Figura III-1).

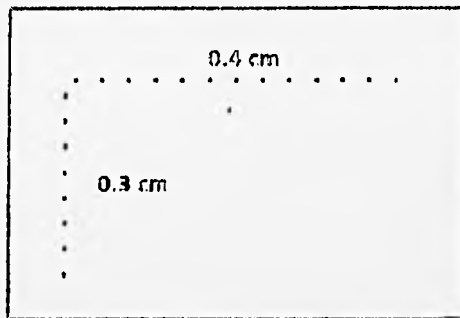


FIGURA III-1
Formato de imagen de 1.33

La razón del ancho horizontal a la longitud vertical es :

$$\frac{0.4}{0.3} = 1.33$$

Esta razón se deriva de que los pixeles no son cuadrados y las pantallas son rectangulares.

La precisión de las imágenes en pantalla depende en cierta medida del formato. El mismo formato se presenta con un círculo: un número igual de pixeles horizontales y verticales resultará una imagen con la apariencia de un huevo (elíptica). Para evitar esta distorsión, siempre que se dibuje una imagen mediante el sistema de coordenadas en pantalla, deberá multiplicarse la altura vertical, como si se tratara de un cuadrado, por el formato de imagen.

III.2 TRAZADO DE PUNTOS

Para dibujar una imagen en una pantalla de barrido, se deben determinar los puntos correspondientes en la memoria de imagen que conforman la imagen. Se deberán, pues, escribir algoritmos de graficación de puntos a base de conversión de barrido.

Tanto la memoria de imagen como la pantalla de barrido, cuentan con un subsistema de coordenadas bidimensional, con el origen en la esquina inferior izquierda. Se accede a cada pixel por medio de un par de coordenadas (x, y) entero no negativo. Los valores de 'x' empiezan en el origen '0', y se incrementan de izquierda a derecha; los valores de 'y' comienzan en '0' y se incrementan de abajo hacia arriba (Figura III-2).

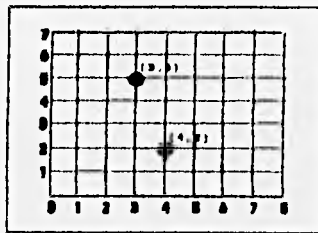


FIGURA III-2
Sistemas de coordenadas de pantallas

Ninguno de los valores coordenados de 'x' o 'y' deben exceder los límites del depósito de presentación. Cualquier intento por representar tales valores puede producir un efecto de **enrollamiento** el punto se "enrolla" y se aparece por el otro lado de la pantalla. Por ejemplo, si se dibuja una línea 'AB', se obtienen las líneas 'AC' y 'DE', como se muestra en la Figura III-3.

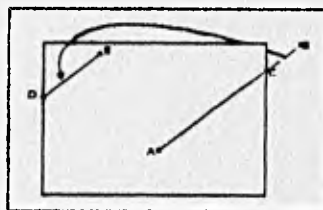


FIGURA III-3
Efecto de enrollamiento

Para prevenir este efecto, los programas de graficación deben probar cada valor, a fin de establecer si el punto representado se encuentra más allá de los límites de la pantalla. Si 'x' es negativo o mayor que el valor de 'x máximo', o 'y' es negativo o mayor que el valor de 'y máximo', debe efectuarse una acción apropiada. Una solución muy sencilla consiste en asignar el valor de 'cero' a cualquier coordenada con valor negativo, o bien, asignar el máximo valor de pantalla a toda coordenada grande que esté fuera del rango.

Para dibujar un punto en la pantalla, se requiere de un procedimiento de graficación de puntos. Se supone aquí la disponibilidad de la orden :

dibujarpunto(x, y)

procedimiento para graficación proporcionada por el sistema, que enciende el pixel ubicado en la posición (x, y) de pantalla (memoria de imagen). Los parámetros 'x' y 'y', deben ser constantes o expresiones enteras no negativas.

La **graficación de un punto**, se instrumenta o implanta en un paquete de gráficas mediante la conversión de la información sobre las coordenadas de un programa de aplicaciones en instrucciones adecuadas para el dispositivo de salida en uso. Con un monitor CRT, por ejemplo, el haz de electrones se enciende para iluminar un punto fosforescente en la localidad especificada en la pantalla. Esto se logra con un despliegue rastreador en blanco y negro fijando el valor del bit en la posición coordenada que se especifica dentro del buffer de cuadros en '1'.

Por lo tanto, mientras el haz de electrones recorre cada línea de rastreo horizontal, este emite un estallido de electrones (traza un punto) siempre que se encuentra un valor de '1' en el buffer de cuadros o de estructura. Para monitores de rastreo al azar, la instrucción de graficación del punto se almacena en el archivo de despliegues y las coordenadas se convierten en deflexiones de tensión que mueven el haz de electrones hacia esa posición durante cada ciclo de renovación.

III.3 TRAZADO DE RECTAS

Muchas imágenes generadas por la computadora se componen de segmentos de rectas. Al encender una cadena de pixeles adyacentes, un segmento aparece desplegado. Para dibujar una recta, es necesario determinar cuales pixeles se hallan cerca de ésta y proporcionar la mejor aproximación a la recta deseada. La Figura III-4, ilustra una recta dibujada en un sistema de presentación por barrido. La exactitud y calidad de la recta dibujada depende de la resolución del dispositivo de presentación. Los sistemas de alta resolución (1024 x 1024) trazan líneas rectas y continuas, que empiezan y terminan con cierta precisión. Los sistemas de baja resolución, por el contrario, dibujan rectas con huecos y con toda clase de saltos (Figura III-5). Las rutinas de trazado de rectas en sistemas gráficos de alta calidad son implantados por un hardware que genera con rapidez los pixeles que conforman la recta, una vez determinados los dos puntos extremos. El trazado de rectas debe ser preciso, eficiente y fácil de implantar.

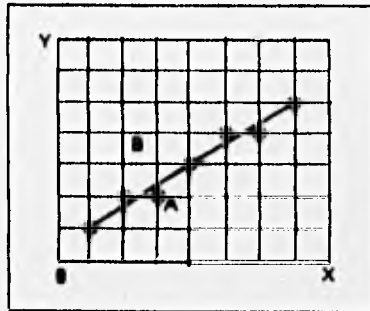


FIGURA III-4
Píxeles que representan una línea recta

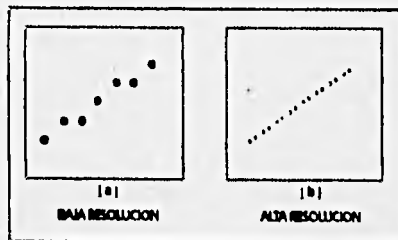


FIGURA III-5
Resolución de pantallas de presentación

Las instrucciones de trazo de líneas de un programa de aplicación definen a las líneas como componentes de una imagen especificando las coordenadas de los puntos extremos de cada línea. El dispositivo de salida recibe órdenes de llenar la trayectoria de la línea entre cada par de puntos extremos. Para dispositivos analógicos, como un graficador de pluma o de despliegue de rastreo al azar con un generador de vectores analógico, se traza de un extremo a otro una línea recta en forma alisada. Las tensiones de deflexión horizontales y verticales que varían linealmente son generadas y proporcionales a los cambios requeridos en las direcciones 'x' y 'y' con el objeto de producir las líneas alisadas.

Los dispositivos digitales, como un despliegue de rastreo con rastreador, produce una línea graficando píxeles entre los dos extremos. Las posiciones de los píxeles se calculan a partir de la ecuación de la recta y los bits adecuados se colocan en el buffer de cuadros. Leyendo el buffer de cuadros, el controlador del despliegue activa después las posiciones correspondientes en la pantalla. Como los píxeles se grafican en posiciones enteras, la línea trazada sólo puede aproximar posiciones de líneas reales entre los puntos extremos especificados. Por ejemplo, si se calcula que la posición (10, 33, 20, 72) está en la línea, se traza la posición del pixel (10, 21). Este redondeo de valores coordenados a enteros ocasiona que se desplieguen líneas con apariencia de escalera ("las dentelladas"), que pueden ser muy notorias en sistemas de resolución inferior. El aspecto de las líneas rastreadoras pueden mejorarse usando sistemas de alta resolución y aplicando también técnicas que se hayan desarrollado específicamente para alisar líneas generadas por puntos.

III.3.1 RECTAS HORIZONTALES Y VERTICALES

Las rectas más fáciles de dibujar son las **rectas verticales y las horizontales**. Las coordenadas en pantalla de los puntos de una recta horizontal, se obtienen manteniendo constante el valor de 'y' e incrementando continuamente el valor de 'x' en una unidad. Para trazar rectas verticales, el valor de 'x' permanece fijo y el valor de 'y' varía. Una recta vertical puede tener huecos visibles cuando es trazada en un sistema de presentación por barrido con baja resolución como resultado del patrón de barrido seguido.

III.3.2 RECTAS DIAGONALES

Para trazar una **recta diagonal** con una pendiente igual a '+1', sólo se necesita incrementar repetidamente en una unidad los valores tanto de 'x' como de 'y' a partir de los pixeles de inicio y de fin. Recuérdese que la pendiente se define como el cambio en los valores de 'y' dividido entre el cambio en los valores de 'x'.

III.3.3 RECTAS ARBITRARIAS

El trazado de **rectas con pendientes arbitrarias** da origen a ciertos problemas. Primero, debido a que las pantallas de presentación pueden iluminarse sólo en posición de pixeles, la presentación por barrido tiene un efecto de escalera que reproduce aproximadamente la recta real (Figura III-4). Aunque podría resultar imposible seleccionar pixeles que coincidan en la recta real, lo importante es encender aquellos que se encuentran muy cerca de ésta. Por ejemplo, en la Figura III-4, el pixel de la posición 'A' es una mejor selección que el de la posición 'B'. Segundo, la determinación de los pixeles más cercanos a la recta no es tarea fácil. Los diferentes algoritmos calculan pixeles distintos para obtener la aproximación a la recta. La selección del algoritmo depende de la velocidad de generación de las rectas, al igual que de la apariencia de éstas. Para comprender mejor esos criterios, observemos diferentes tipos de algoritmos.

III.4 ALGORITMOS PARA TRAZOS DE LINEAS

La ecuación de una recta puede enunciarse en la forma :

$$y = m \cdot x + b \quad 3-1$$

con 'm' como pendiente de la recta, y 'b' como intersección 'y'. Dado que los dos extremos de un segmento rectilíneo se especifican como (x_1, y_1) y (x_2, y_2) , como se muestra en la Figura III-6,

podemos determinar valores de la pendiente 'm' y de la intersección 'y', 'b', con los siguientes cálculos:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad 3-2$$

$$b = y_1 - m \cdot x_1 \quad 3-3$$

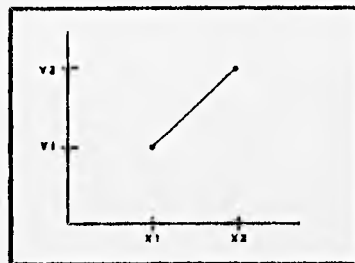


FIGURA III-6
Segmento de línea, especificado por los extremos coordenados (x_1, y_1) y (x_2, y_2)

Los algoritmos para desplegar líneas rectas se basan en la ecuación de la recta '3-1' y los cálculos que se dan en las ecuaciones '3-2' y '3-3'.

Para cualquier intervalo ' Δx ' de 'x' a lo largo de una recta, para calcular el intervalo ' Δy ' de 'y' correspondiente a la ecuación '3-2' como :

$$\Delta y = m \cdot \Delta x \quad 3-4$$

Esta ecuación forma la base para determinar tensiones de deflexión en dispositivos analógicos. La variación en la tensión de deflexión horizontal se hace proporcional ' Δx ' y el cambio en la tensión de deflexión vertical se hace proporcional al valor de ' Δy ' calculado a partir de la ecuación '3-4'. Estas deflexiones se usan después para generar una línea con pendiente 'm' entre los extremos que se especifican.

III.4.1 ALGORITMO DDA

El **Analizador Diferencial Digital (DDA)** es un algoritmo para calcular las posiciones de los pixeles a lo largo de una línea, mediante el uso de la ecuación '3-4'. Esto se lleva a cabo realizando pasos unitarios con una coordenada 'y' calculando los valores correspondientes a la otra coordenada.

Primero consideremos una línea con pendiente positiva, como se muestra en la Figura III-6. Si la pendiente es menor que o igual a '1', se toma la variación en los valores de la coordenada 'x' como '1' y se calcula cada valor sucesivo de la coordenada 'y' como :

$$y_{i+1} = y_i + m \quad 3-5$$

El subíndice 'i' toma valores enteros comenzando desde '1', para el primer punto, y aumenta de '1' en '1' hasta que se alcanza el extremo final. Como 'm' puede ser un número real cualquiera, los valores calculados de 'y' deben redondearse al entero más próximo.

Para rectas con una pendiente positiva mayor que '1', se invierten los papeles de 'x' y de 'y'. Es decir, nos movemos en etapas de 'y' unitarias y calculamos cada valor sucesivo de 'x' como :

$$x_{i+1} = x_i + \frac{1}{m} \quad 3-6$$

Las ecuaciones '3-5' y '3-6' suponen que procedemos a lo largo de la línea desde el extremo izquierdo hasta el derecho (Figura III-1). Si estos puntos extremos se invierten de manera que el punto extremo inicial esté del lado derecho, entonces tenemos ' $\Delta x = -1$ '.

$$y_{i+1} = y_i - m \quad 3-7$$

o bien (cuando la pendiente es mayor que '1') se tiene ' $\Delta y = -1$ ' con :

$$x_{i+1} = x_i - \frac{1}{m} \quad 3-8$$

Las ecuaciones '3-5' a '3-8' también pueden ser utilizadas para calcular puntos a largo de una recta como pendiente negativa. Si el valor absoluto de la pendiente es menor que '1' y el punto extremo inicial esta a la izquierda se hace ' $\Delta x = 1$ ' y se calculan valores de 'y' con la ecuación '3-5'. Cuando el punto extremo inicial está a la derecha (con la misma pendiente) se hace ' $\Delta x = -1$ ' y se obtienen posiciones de 'y' a partir de la ecuación '3-7'.

Análogamente, cuando el valor absoluto de una pendiente negativa es mayor que '1', se utiliza ' $\Delta y = -1$ ' y la ecuación '3-8' o bien, se usa ' $\Delta y = 1$ ' y la ecuación '3-6'.

Este algoritmo se resume en el siguiente procedimiento, en el cual se aceptan como entrada los puntos extremos de la recta (x_1, y_1) y (x_2, y_2) . Las diferencias en los valores de las coordenadas de entrada en cada dirección se calculan como parámetros 'dx' y 'dy'. La diferencia con la magnitud mayor determina el valor del parámetro steps, que especifica el número de puntos que se graficarán a lo largo de la recta.

Comenzando en la posición (x_1, y_1) , se suma una cantidad a cada coordenada para generar la siguiente posición. Esto se repite step veces. Si la magnitud de 'dx' es mayor que la magnitud de 'dy' y ' x_1 ' es menor que ' x_2 ', los valores de los incrementos en la dirección 'x' y 'y' son '1' y 'm', respectivamente. Si el cambio mayor se observa en la dirección 'x', pero ' x_1 ' es mayor que ' x_2 ', entonces se suman los valores de '-1' y '-m' para generar cada nuevo punto sobre la recta. De lo contrario, se utiliza un incremento (o decremento) de 'x' de $\frac{1}{m}$.

Se supone que los puntos se trazarán en un sistema de una sola intensidad de manera que el comando `set_pixel` sea una llamada del procedimiento para almacenar un valor de pixel de '1' ("encendido") en el buffer de cuadros en una posición especificada por los parámetros coordenados 'x' y 'y'.

El Algoritmo DDA es el siguiente :

```

procedure dda (x1, y1, x2, y2 : Integer),
  var
    dx, dy, steps, k : Integer;
    x_increment, y_increment, x, y : real;

  begin
    dx := x2 - x1;
    dy := y2 - y1;
    if abs(dx) > abs(dy) then steps := abs(dx)
      else steps := abs(dy);
    x_increment := dx / steps;
    y_increment := dy / steps;
    x := x1; y := y1;
    set pixel (round(x), round(y));
    for k := 1 to steps do begin
      x := x + x_increment;
      y := y + y_increment;
      set_pixel (round(x), round(y))
    end {for k}
  end; {dda}

```

El Algoritmo DDA es un método más rápido para calcular posiciones del pixel que el uso directo de la ecuación '3-1'. Elimina la multiplicación en la ecuación '3-1' aprovechando las características del rastreador, de manera que las etapas unitarias se tomen en la dirección 'x' o 'y' hacia la siguiente posición del pixel a lo largo de la recta.

Sin embargo, los cálculos disminuyen en rapidez por las divisiones que se necesitan para fijar valores de incremento, el uso de la aritmética de punto flotante y las operaciones de redondeo.

III.4.2 ALGORITMO DE LA LINEA DE BRESENHAM

Un algoritmo de línea más efectivo para determinar las posiciones del pixel, creado por Bresenham, haya las coordenadas enteras más próximas a la trayectoria real de la recta utilizando solamente aritmética entera. La Figura III-7 ilustra secciones de una pantalla de despliegue donde se trazarán segmentos rectilíneos. Las posiciones de los pixeles en la pantalla se representan por las áreas rectangulares situadas entre líneas de una retícula. En cada uno de estos ejemplos, se necesita escoger entre dos alternativas de pixel en cada posición 'x', comenzando desde el punto extremo izquierdo de la recta en la Figura III-7, se necesita determinar el siguiente punto de la recta donde se trazará la posición (11, 10) o bien (11, 11). Por lo tanto, el siguiente pixel que se grafique será aquel cuyo valor de 'y' esté más próximo a la posición real de 'y' sobre la recta.



FIGURA III-7
Sección de una pantalla donde se despliega un segmento rectilíneo comenzando desde la posición (10, 10). Las posiciones de los pixeles se representan por las áreas rectangulares numeradas

Empezaremos con una recta cuya pendiente es positiva y menor que '1'. Las posiciones de los pixeles a lo largo de la trayectoria de la línea pueden trazarse después tomando etapas unitarias en la dirección 'x' determinando el valor de la coordenada 'y' del pixel más cercano a la recta 'm' cada etapa. Para establecer los cálculos que se necesitan en el algoritmo, se considera la situación que se muestra en la Figura III-8.

En esta figura se supone que la posición del pixel (x_i, y_i) se ha trazado y ahora se necesita decidir cuál es el siguiente pixel que se graficará. Las dos alternativas de la siguiente posición del pixel están en las coordenadas $(x_i + 1, y_i)$ y $(x_i + 1, y_i + 1)$.

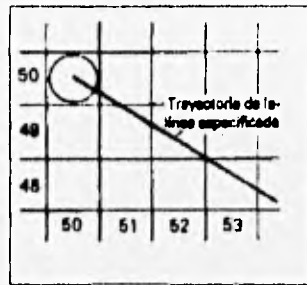


FIGURA III-8
Selección de una retícula de la pantalla donde se desplegará una línea que pasa por (x_i, y_i)

En la Figura III-9, las diferencias de coordenadas entre el centro de los dos pixeles y la coordenada 'y' de la recta se rotulan 'd₁' y 'd₂'. La posición 'y' puede calcularse como :

$$y = m(x_i + 1) + b$$

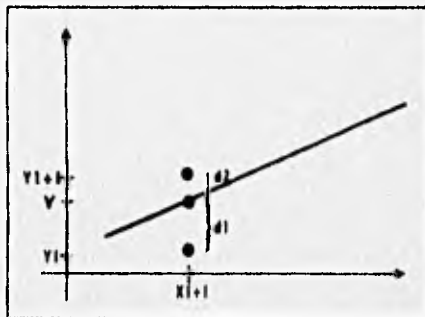


FIGURA III-9
Diferencias coordenadas entre los centros de pixeles y la posición sobre la trayectoria de la línea en $x_i + 1$

Por lo tanto,

$$d_1 = y - y_i$$

$$d_1 = m(x_i + l) + b - y_i$$

$$d_2 = (y_i + l) - y$$

$$d_2 = y_i + l - m(x_i + l) - b$$

La diferencia entre estas dos distancias es :

$$d_1 - d_2 = 2m(x_i + l) - 2y_i + 2b - l \quad 3-9$$

Ahora se definirá un parámetro que ofrece una medida de las distancias relativas de dos pixeles de la posición actual sobre una recta dada. Sustituyendo $m = \frac{\Delta y}{\Delta x}$, en la ecuación '3-9', tenemos que :

$$p_i = \Delta x(d_1 - d_2) \quad 3-10$$

$$p_i = 2\Delta y \cdot x_i - 2\Delta x \cdot y_i + c$$

La constante 'c' tiene el valor $2\Delta y + \Delta x(2b - l)$ y se podría calcular una sola vez para todos los puntos, pero se observará que la ecuación '3-10' puede ser revisada a fin de eliminar esta constante. El parámetro 'p_i' tiene un valor negativo, si el pixel en la posición 'y_i' está más próximo a la recta que el pixel superior. En este caso, se seleccionará el pixel inferior; de lo contrario, se elige el pixel superior.

La ecuación '3-10' se simplifica relacionando parámetros de intervalos sucesivos de 'x'. Por lo tanto, el valor de cada parámetro sucesivo se obtiene a partir del parámetro calculado con anterioridad. Podemos describir la ecuación '3-10' en la forma :

$$p_{i+1} = 2\Delta y \cdot x_{i+1} - 2\Delta x \cdot y_{i+1} + c$$

Restando la ecuación '3-10' se tiene :

$$p_{i+1} - p_i = 2\Delta y(x_{i+1} - x_i) - 2\Delta x(y_{i+1} - y_i)$$

Pero $x_{i+1} = x_i + 1$, de modo que :

$$p_{i+1} = p_i + 2\Delta y - 2\Delta x(y_{i+1} - y_i) \quad 3-11$$

Esta ecuación nos da una manera de calcular el valor de cada parámetro sucesivo a partir del anterior. El primer parámetro, 'p₁' se obtiene de la evaluación de la ecuación con (x₁, y₁) como punto extremo inicial y $m = \frac{\Delta y}{\Delta x}$:

$$p_1 = 2\Delta y - \Delta x \quad 3-12$$

Las etapas del **algoritmo de Bresenham** se resumen en el Cuadro III-1, para una recta con pendiente positiva menor que '1'. Como las constantes '2Δy', 'Δx' y '2(Δy-Δx)' necesitan ser evaluadas y almacenadas sólo una vez, la aritmética comprende únicamente la adición y la sustracción de enteros. Un procedimiento para implantar el algoritmo del Cuadro III-1 se da en el siguiente programa. Las coordenadas de los puntos extremos de la recta sirven de entrada para este procedimiento a través de los parámetros 'x₁', 'y₁', 'x₂' y 'y₂'. La llamada de **set_pixel** fija la posición en el buffer de estructura para el punto seleccionado.

El **Algoritmo de Bresenham** es el siguiente :

```

procedure bres_line (x1, y1, x2, y2 : integer);
  var
    dx, dy, x, y, x_end, p const1, const2 : integer;
  begin
    dx := abs(x1 - x2);
    dy := abs(y1 - y2);

```

```

p := 2 * dy - dx;
const1 := 2 * dy;
const2 := 2 * (dy - dx);
{determine que punto usar como inicial, y cual como final}
if x1 > x2 then begin
  x := x2;
  y := y2;
  x_end := x1;
end {if x1 > x2}
else begin
  x := x1;
  y := y1;
  x_end := x2;
end; {if x1 <= x2}
set_pixel pixel (x, y);
while x < x_end do begin
  x := x + 1;
  if p < 0 then p := p + const1
  else begin
    y := y + 1
    p := p + const2
  end; {else begin}
  set_pixel pixel (x, y)
end {while x < x_end}
end; {bres_line}

```

Hasta ahora ha sido limitado el estudio de las rectas con pendiente positiva entre '0' y '1'. Podemos ampliar el algoritmo a pendientes positivas mayores que '1' intercambiando los papeles de las coordenadas 'x' y 'y'. Es decir, nos dirigimos por la dirección 'y' en etapas unitarias y calculamos posiciones sucesivas de 'x'. En el caso de pendientes negativas; los procedimientos son similares, excepto que ahora una coordenada decrece mientras que la otra crece.

Algoritmo De Bresenham.

1. Dar como entrada los extremos de la linea. Posteriormente almacenar el punto de extremo izquierdo en (x_1, y_1) . Por último, almacenar el extremo derecho en (x_2, y_2) .
2. El primer punto que se seleccionará.
3. Calcular ' $\Delta x = x_2 - x_1$ ', ' $\Delta y = y_2 - y_1$ ' y ' $p_1 = 2\Delta y - \Delta x$ '. Si ' $p_1 = 0$ ', el siguiente punto que se fijará es $(x_1 + 1, y_1)$. En caso contrario, el siguiente punto es $(x_1 + 1, y_1 + 1)$.
4. Continuar incrementando la coordenada 'x' en pasos unitarios. En la posición ' $x_i + 1$ ', la coordenada que se seleccionará ' $y_i + 1$ ', es ' y_i ' o bien, ' $y_i + 1$ ', según ' $p_i < 0$ ' o bien ' $p_i > 0$ '. Los cálculos de cada parámetro 'p' dependen del último. si ' $p_i < 0$ ', la forma del siguiente parámetro es :

$$p_{i+1} = p_i + 2\Delta y$$

Pero si ' $p_i \geq 0$ ', el siguiente parámetro es :

$$p_{i+1} = p_i + (2\Delta y - \Delta x)$$

Por lo tanto, si ' $p_i + 1 < 0$ ', la siguiente coordenada 'y' que se seleccionará es ' $y_i + 1$ '. En caso contrario, seleccionar ' $y_i + 1 + 1$ '. (La coordenada ' $y_i + 1$ ' se determinó como ' y_i ', o bien ' $y_i + 1$ ' por medio del parámetro ' p_i ' del paso 3).

5. Repetir los procedimientos del paso 4 hasta que la coordenada 'x' llegue a ' x_2 '.

CUADRO III-1
Algoritmo de Bresenham

III.4.3 CARGAR EL BUFFER DE ESTRUCTURA

Siempre que se van a desplegar puntos y líneas con un sistema rastreador, el contenido del buffer de estructura debe ser modificado para que contenga valores de intensidad adecuados para las posiciones coordenadas que se especifican. Se ha supuesto que esto se lleva a cabo con el procedimiento `set_pixel`. Este procedimiento convierte valores de coordenadas en direcciones correspondientes dentro del rastreador y almacena valores de intensidad en estas posiciones en el arreglo del buffer de estructura.

Como ejemplo específico, supóngase que el arreglo del buffer de estructura se direcciona en orden del renglón principal en un monitor de despliegue con localidades coordenadas que varían de (0, 0) en la esquina inferior izquierda a (x_{max} , y_{max}) en la esquina superior derecha. Con un monitor de dos niveles (que requiere un bit de almacenamiento por pixel), la dirección del bit dentro del rastreador de una posición coordenada en la pantalla (x, y) se calcula así :

$$ADDR(x,y) = ADDR(0,0) + y(x_{max} + 1) + x \quad 3-13$$

Este cálculo se instrumenta en el procedimiento `set_pixel` cuando debe trazarse un sólo punto y cuando el valor de intensidad del extremo inicial de una recta debe ser fijado. Para un algoritmo de trazos de líneas, se pueden simplificar los cálculos de la ecuación '3-13' para puntos intermedios a lo largo de la recta tomando ventaja del hecho que se realizan etapas unitarias en las direcciones (x, y). El cálculo de direcciones dentro del buffer de estructura de estos puntos puede realizarse incrementando previamente las direcciones calculadas. Por ejemplo, desde cualquier posición (x, y), la siguiente dirección que se cargará en el rastreador (de una recta con pendiente positiva menor que '1' puede calcularse como una de las dos posibilidades siguientes :

$$ADDR(x + 1, y) = ADDR(x, y) + 1 \quad 3-14$$

$$ADDR(x + 1, y + 1) = ADDR(x, y) + x_{max} + 2 \quad 3-15$$

Pueden obtenerse cálculos recursivos semejantes de otros cambios de coordenadas a partir de la ecuación 3-13. Estas relaciones ofrecen un método efectivo de direccionamiento ya que en los cálculos sólo intervienen la adición de enteros.

Los métodos para almacenar el buffer de estructura e implantar el procedimiento `set_pixel` dependen de las capacidades de un sistema determinado y de los requisitos de diseño del paquete de software con sistemas que hacen uso del color o de un intervalo de valores de intensidad, un parámetro adicional de entrada de `set_pixel` especificará el valor de intensidad que se almacena en el buffer de estructura.

III.4.4 LINEAS CON ANTISEUDONIMOS

Los algoritmos rastreadores que se han analizado hasta ahora generan líneas que tienen una apariencia de escalinata o escalera. Pueden utilizarse resoluciones mayores para mejorar el aspecto de las líneas, pero esto requiere de buffers de renovación de mayor tamaño y este punto de vista no elimina o suprime por completo el efecto de escalera. Las representaciones de objetos trazados en un rastreador están sujetas a distorsiones debido a los **seudónimos**. El proceso de digitalización hace girar puntos coordinados sobre el objeto para disimular las posiciones de los píxeles enteros en rastreador. Podemos modificar algoritmos de trazos de líneas para compensar este efecto del rastreador añadiendo rutinas con **antiseudónimos** que suavicen el despliegue de una línea en un monitor de video. Las técnicas de tratamiento con antiseudónimos suprimen el aspecto de escalera ajustando las intensidades de los píxeles a lo largo de la trayectoria de la recta.

Un método para crear una rutina con antiseudónimos se basa en la **Teoría del Muestreo**. La idea de este método es que las entidades geométricas naturales, como los puntos y líneas trazados en una pantalla de despliegue, tienen dimensiones finitas. Un pixel no es un punto matemático infinitesimal sino una mancha de luz que cubre una área pequeña de una pantalla. Y las líneas tienen una anchura aproximadamente igual a la de un pixel. Cuando se digitalizan objetos naturales en retícula rectangular (rastreador), las áreas de la retícula son "muestreadas" para

determinar valores adecuados de alta intensidad. Los sistemas rastreadores que pueden desplegar más de dos niveles de intensidad pueden usar este método para ajustar pixeles de manera que cada área de la retícula tenga la intensidad adecuada.

La Figura III-10 muestra una línea representada con una anchura finita en una retícula de pixeles. Las áreas de los pixeles se suponen cuadradas, y la anchura de la línea se hace igual a la anchura de un pixel. En vez de trazar la línea con un sólo pixel en cada posición de 'x', todos los pixeles que están debajo del área de la línea se despliegan con una intensidad proporcional al área de cobertura. En el ejemplo que se muestra, los pixeles situados en posiciones (11, 10) y (11, 11) están cubiertos cerca de la mitad por la línea. Así, cada uno de estos pixeles se coloca en un nivel de intensidad de aproximadamente 50% del máximo. En forma análoga, el pixel en la localidad (9, 10) se fija en una intensidad cerca del 10% del máximo. Aunque este método de antiseudónimos puede mejorar el aspecto de las líneas, los cálculos requieren mucho tiempo.

El ajuste de intensidades de los pixeles a lo largo de una recta también compensa el efecto de otro rastreador, que se ilustra en la Figura III-11. Ambas líneas se trazan con el mismo número de pixeles, no obstante que la línea diagonal es más larga que la horizontal en un factor (raíz cuadrada de 2).

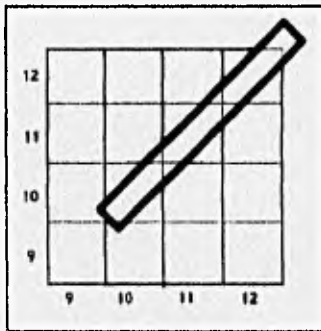


FIGURA III-10
Representación de una línea como un rectángulo de anchura finita en una retícula de pixeles. El punto extremo izquierdo de la línea está en la posición (10, 10)

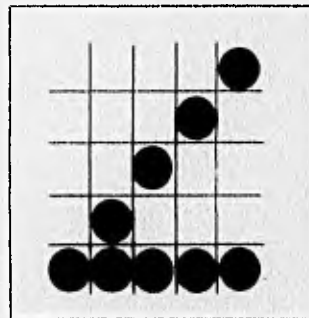


FIGURA III-11
Líneas de longitud desigual desplegadas con el mismo número de pixeles en cada línea

El efecto visual de esto es que la línea diagonal parece menos brillante que la horizontal, ya que la línea diagonal se despliega con una intensidad menor por longitud unitaria. Un algoritmo de trazo de líneas podría adaptarse para compensar este efecto mediante el ajuste de intensidad de cada línea según su pendiente. Las líneas horizontales y verticales se desplegarán con la misma intensidad, mientras que las líneas de 45° tendrán la mayor intensidad. Sin embargo, si se aplican técnicas de antiseudónimos a un despliegue, las intensidades se compensan automáticamente. Cuando la anchura finita de las líneas se toma en cuenta, las intensidades de los píxeles se ajustan de manera que las líneas desplieguen una intensidad total proporcional a su longitud.

Otra técnica de antiseudónimos es el punto de vista del **Ajuste de Fase de Píxeles**, creado por Megatek Corporation. Las intensidades de las aristas de las líneas se ajustarán por la "microposición" del haz de electrones. Los sistemas que incorporan esta técnica están diseñados de manera que las posiciones individuales de los píxeles puedan cambiarse en una fracción de diámetro de un píxel. Admitiendo cambios de $\frac{1}{4}$, $\frac{1}{2}$ y $\frac{3}{4}$ de diámetro de un píxel, una línea puede ser desplegada trazando puntos más próximos a la trayectoria real de la línea. Estos sistemas también permiten modificar el tamaño de píxeles individuales. La Figura III-12 ilustra la colocación de antiseudónimos en la línea mediante el uso de la técnica de ajuste de fase de los píxeles.

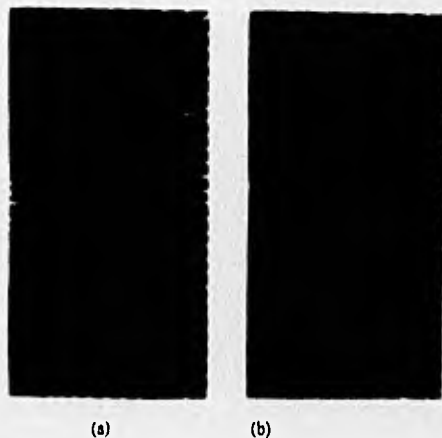


FIGURA III-12
Líneas en escalera (a) graficadas en el sistema Merlin 9200, son aisladas (b) con una técnica de antiseudónimo llamada ajuste de fase de píxeles. Esta técnica incrementa el número de puntos direccionables en el sistema de 768 x 576 a 3072 x 2304

III.4.5 COMANDOS DE LINEAS

Los paquetes de gráficas pueden diseñarse para incluir un comando para el trazo básico de líneas y otro para graficar puntos individuales. Muchas aplicaciones de gráficas implican la construcción de figuras con segmentos rectilíneos. Además, un sólo comando puede servir para ambos fines, ya que un punto puede considerarse como un segmento de recta muy corto.

Un comando para graficar ambos puntos y líneas puede definirse en la forma :

polyline (n, x, y)

Este comando se utiliza para especificar un sólo punto, un sólo segmento rectilíneo o bien, una serie de segmentos de líneas conectadas, según el valor asignado al parámetro 'n'. Los valores coordenados de los extremos de la línea (o del punto sólo) se almacena en los arreglos 'x' y 'y'.

La implantación de la función **polyline** se hace a través de los procedimientos **set_pixel** y de trazo de líneas, como el algoritmo de Bresenham. Cuando debe tratarse un sólo punto, se invoca la rutina **set_pixel**. Cuando se va a trazar una línea o bien, una serie de ellas, la función **polyline** ocasiona que se ejecute el procedimiento de trazo de líneas.

Para generar puntos, se fijará el valor de 'n' en '1' y se dará los valores coordenados de (x, y) en 'x[1]' y 'y[1]'. Para poner un ejemplo, las proposiciones son:

```
x[1] := 150;
y[1] := 100;
polyline (1, x, y);
```

las cuales especifican que un sólo punto se graficará en las posiciones coordenadas (150, 100). Se supone que las referencias coordenadas en el comando **polyline** se enuncian como valores de coordenadas absolutas. Esto significa que los valores especificados son las posiciones reales de puntos en el sistema coordenado en uso.

Algunos sistemas de gráficas utilizan comandos de líneas (y puntos) con especificaciones de coordenadas relativas. En este caso, los valores de coordenadas se expresan como desplazamientos desde la última posición referida (llamada posición actual). Por ejemplo, si la localización (3, 2) es la última posición de la cual se ha hecho referencia en un programa de aplicación, la especificación de una coordenada relativa de (2, -1) corresponde a una posición absoluta de (5, 1).

Se obtiene un segmento rectilíneo con un comando como **polyline** haciendo 'n' igual a '2' y asignando valores de los dos extremos de coordenadas de la recta a los arreglos 'x' y 'y'. El ejemplo que sigue es un segmento de programa que especifica una línea con extremos en (50, 100) y (250, 25).

```
x[1] := 50;
y[1] := 100;
x[2] := 250;
y[2] := 25;
polyline (2, x, y);
```

Con paquetes de gráficas que emplean el concepto de posición actual, un usuario sólo necesita dar un punto coordinado en una proposición de trazo de líneas. Esto indica al sistema desplegar una línea de la posición actual a las coordenadas dadas. La posición actual se actualiza después a la localización de coordenadas que se expresó en el comando de líneas. Una serie de líneas conectadas se produce con estos paquetes mediante una secuencia de comandos de líneas, uno por cada línea que se trazará.

Como puede especificarse cualquier número de puntos con la función **polyline**, una persona tiene la capacidad de generar una secuencia de segmentos de líneas conectados con una proposición. Esto se hace haciendo igual a 'n' el número de extremos de líneas y almacenando las coordenadas del extremo en los arreglos 'x' y 'y'. El paquete de gráficas despliega después de una serie de 'n - 1' segmentos de líneas que conectan las 'n' coordenadas adyacentes de la posición (x[1], y[1]) a la posición (x[n], y[n]).

Podemos implantar el comando `polyline` para ' $n \geq 2$ ' con un procedimiento que haga llamadas repetidas del algoritmo de trazo de líneas. Cada llamada sucesiva del algoritmo de trazo de líneas pasa el par coordenado que se necesita para graficar el siguiente segmento de línea. El algoritmo de trazo de líneas es accesado por este procedimiento un total de ' $n - 1$ ' veces.

III.4.6 LLENADO DE AREAS

Cuando van aplicarse modelos de sombreados de color a áreas de una escena o gráfica, conviene al usuario poder especificar el área que se llenará. Aunque los modelos de llenado podrían aplicarse a los métodos anteriores de los límites de un polígono definido con un comando de líneas, el procesamiento se simplifica si se utiliza un procedimiento aparte para definir el llenado de un área.

Este punto de vista permite que un área designada se señale de inmediato como aquella que se desplegará con un interior especificado.

Si se presenta el comando siguiente para definir el llenado del área de un polígono:

fill_area (n, x, y)

El área que se llenará está dentro del límite o frontera definido por la serie de ' n ' segmentos de líneas conectados de $(x[1], y[1])$ a $(x[n], y[n])$ y de regreso a $(x[1], y[1])$.

La implantación del comando `fill_area` de un paquete de gráficas depende del tipo de llenado que se use para desplegar el área. Un usuario podría desear que el área quedara en blanco o bien, que se llenara con un color sólido. Cuando el interior del área se deja en blanco, `fill_area` simplemente produce el perfil de la frontera o límite.

Esto es análogo a utilizar el procedimiento `polyline` con las coordenadas iniciales y finales fijadas en los mismos valores.

III.5 ALGORITMOS DE GENERACION DE CIRCUNFERENCIAS

Los parámetros básicos que definen una **circunferencia** son las coordenadas del centro (x_c, y_c) y el radio ' r '. Podemos expresar la ecuación de una circunferencia en varias formas, mediante parámetros de coordenadas cartesianas o polares. La Figura III-13 muestra la relación existente entre los parámetros cartesianos y los polares.

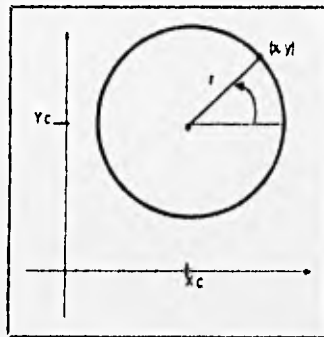


FIGURA III-13
Relación entre coordenadas cartesianas y polares

III.5.1 ECUACIONES DE CIRCUNFERENCIAS

Una forma estándar de la ecuación de la circunferencia es el **Teorema de Pitágoras** :

$$(x - x_c)^2 + (y - y_c)^2 = r^2 \quad 3-16$$

Estas ecuaciones podrían usarse para trazar una circunferencia recorriendo el eje ' x ' en pasos unitarios de ' $x_c - r$ ' a ' $x_c + r$ ' y calculando valores de ' y ' correspondientes en cada posición como :

$$y = y_c \pm \sqrt{r^2 - (x - x_c)^2} \quad 3-17$$

Obviamente, este punto de vista implica una tarea de cálculo considerable en cada etapa y el espaciamiento entre las posiciones de los pixeles trazados no es uniforme, como se muestra en la Figura III-14. Podríamos ajustar el espaciamiento intercambiando 'x' y 'y' (probando valores de 'y' y calculando los valores de 'x') siempre que el valor absoluto de la pendiente de la circunferencia se hiciera mayor que '1'. Pero esto agrega cálculos y la verificación del algoritmo. Una manera de eliminar el espaciamiento desigual asociado con la ecuación '3-17' consiste en calcular puntos situados en la frontera circular mediante el uso de coordenadas polares. Si se expresa la ecuación de la circunferencia en forma polar paramétrica se obtiene el par de ecuaciones :

$$\begin{aligned}x &= xc + r \cdot \cos \theta \\y &= yc + r \cdot \sen \theta\end{aligned}\quad 3-18$$

Cuando un despliegue se genera con estas ecuaciones mediante el uso de un valor angular fijo de ' θ ', se traza una circunferencia con puntos igualmente espaciados a lo largo de la circunferencia. El tamaño de la etapa seleccionada de ' θ ' depende de la aplicación. Para generar circunferencias con un sistema de rastreo con rastreador, se puede fijar el tamaño de la etapa en ' $1/r$ ' y calcular posiciones de pixeles con muy poco espacio entre ellas. Este tamaño de etapa nos da pixeles separados aproximadamente una unidad.

Pueden mejorarse estos métodos aprovechando la simetría de las circunferencias. Un punto dado de la circunferencia puede trazarse en otros puntos de la circunferencia intercambiando coordenadas y alterando el signo de los valores coordenados. Como se ilustra la Figura III-15, un punto en la posición (x, y) en un sector octavo de una circunferencia puede utilizarse para graficar los otros siete puntos que se muestran. Mediante el uso de este enfoque, podrían generarse todas las posiciones de pixeles alrededor de una circunferencia de un despliegue con rastreador calculando solamente los puntos dentro del sector de ' $x = 0$ ' a ' $x = y$ '.

La determinación de las posiciones de pixeles a lo largo de una circunferencia mediante el uso de la ecuación '3-17' o '3-18' requiere una cantidad considerable de tiempo de cómputo. El enfoque del Teorema de Pitágoras implica multiplicaciones y extracciones de raíces cuadradas, mientras

que las ecuaciones paramétricas contienen cálculos trigonométricos y multiplicaciones. Podemos depurar la eficiencia de la generación de circunferencias aplicando un método que reduce los cálculos lo más posible a aritmética entera.

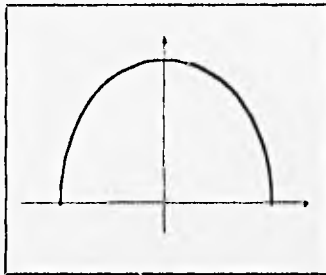


FIGURA III-14
Semicircunferencia positiva trazada con la ecuación $x^2 + y^2 = r^2$ y $(x_c, y_c) = (0, 0)$

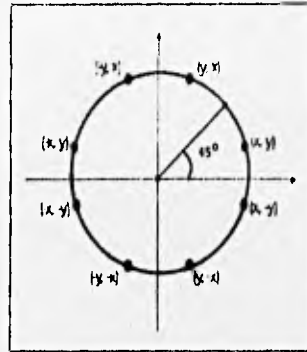


FIGURA III-15
Simetría de una circunferencia. El cálculo de un punto (x, y) situado en el segmento de la circunferencia del primer octante también produce los siete puntos adicionales que se muestran en la circunferencia.

III.5.2 ALGORITMO DE CIRCUNFERENCIA DE BRESENHAM

En el **Algoritmo de Circunferencias de Bresenham**; como sucede en el algoritmo de generación de líneas, las posiciones enteras a lo largo de una trayectoria circular puede obtenerse determinando cuál de los dos pixeles está más próximo a la circunferencia en cada etapa. Para simplificar los enunciados del algoritmo, primero se considera una circunferencia con centro en el origen coordinado ($x_c = 0$ y $y_c = 0$). También se calculan los puntos de un octavo de segmento de una circunferencia suponiendo que se obtendrán los puntos restantes por simetría para almacenarse en un rastreador. (Un sistema de rastreo al azar con un generador de vector podría ampliar los cálculos a través de un ciclo completo.) Se toman etapas unitarias en el sentido 'x', comenzando desde 'x = 0', y terminado cuando 'x = y'. La coordenada inicial de nuestro algoritmo es por tanto, $(0, r)$.

En la Figura III-16 se muestra la situación en alguna etapa arbitraria del algoritmo. Se supone que la posición (x_i, y_i) se ha determinado como más próxima a la trayectoria de la circunferencia. La siguiente posición es por tanto $(x_i + 1, y_i)$ o bien $(x_i + 1, y_i - 1)$.

Según la ecuación '3-16', el valor real de 'y' en la trayectoria de la circunferencia se determina como :

$$y^2 = r^2 - (x_i + 1)^2$$

La Figura III-17 ilustra la relación entre 'y' y los valores coordenados enteros, 'y_i' y 'y_i - 1'. Una medida de la diferencia en las posiciones coordenadas pueden definirse en términos del cuadrado de los valores de 'y' como :

$$\begin{aligned} d_1 &= y_i^2 - y^2 \\ d_2 &= y_i^2 - r^2 + (x_i + 1)^2 \end{aligned} \quad 3-19$$

y

$$\begin{aligned} d_2 &= y^2 - (y_i - 1)^2 \\ d_2 &= r^2 - (x_i + 1)^2 - (y_i - 1)^2 \end{aligned} \quad 3-20$$

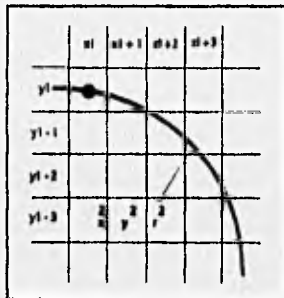


FIGURA III-16
Sección de una retícula de la pantalla donde se desplegará una circunferencia que pasa por el punto (x_i, y_i)

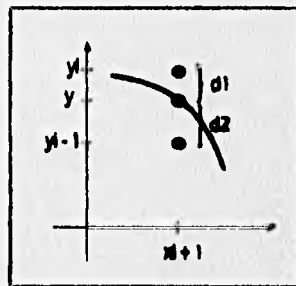


FIGURA III-17
Diferencias coordenadas entre los centros de los píxeles y la posición 'y' en una circunferencia en 'x_i + 1'

Ahora se crea un parámetro para determinar la siguiente posición coordinada como la diferencia entre 'd₁' y 'd₂'.

$$p_i = d_1 - d_2$$

$$p_i = 2(x_i + 1)^2 + y_i^2 + (y_i - 1)^2 - 2r^2 \quad 3-21$$

Si 'p_i' es negativa, se selecciona el pixel en la posición 'y_i'. De lo contrario, se selecciona el pixel situado en la localidad 'y_i - 1'.

La prueba de la selección del siguiente pixel se cumple si la trayectoria real pasa sobre 'y_i', o bien, debajo de 'y_i - 1', como se muestra en la Figura III-18 (a), se tiene 'd₁ < 0', 'd₂ < 0' y 'p_i < 0', de manera que el punto en 'y_i' sería el seleccionado. En el segundo caso, la Figura III-18 (b), 'd₁ > 0', 'd₂ < 0'. Ahora 'p_i > 0' y el punto en 'y_i - 1' es el seleccionado.

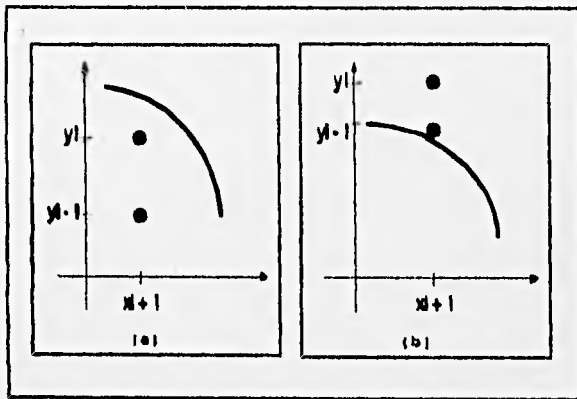


FIGURA III-18
Posibles posiciones de los pixeles
(a) ambos centros de los pixeles
están debajo de la trayectoria de
la circunferencia y (b) ambos
centros de los pixeles están sobre
la trayectoria de la circunferencia

Una forma recursiva del parámetro 'p' se obtiene evaluando 'p_i + 1' en términos de 'p_i' :

$$p_{i+1} = 2[(x_i + 1) + 1]^2 + y_{i+1}^2 + (y_{i+1} - 1)^2 - 2r^2$$

Esta expresión puede escribirse en términos de la ecuación '3-21' :

$$p_{i+1} = p_i + 4x_i + 6 + 2(y_{i+1}^2 - y_i^2) - 2(y_{i+1} - y_i) \quad 3-22$$

La posición ' y ', ' $y_i + 1$ ' es la misma que ' y ', o bien, la misma que ' $y_i - 1$ ', según el valor de ' p_i '. Comenzando desde ' p_1 ', el algoritmo determina cada parámetro ' p ' sucesivo desde o a partir del anterior.

Se obtiene ' p_1 ' haciendo $(x_1, y_1) = (0, r)$ en la ecuación '3-21' :

$$p_1 = 3 - 2r \quad 3-23$$

El Cuadro III-2 resume las etapas que se llevan para calcular coordenadas enteras lo más próximas a la circunferencia definida.

Para generalizar el algoritmo de manera que pueda trazarse una circunferencia con posición central arbitraria, simplemente se agrega ' x_c ' a cada valor sucesivo de ' x ' y se agrega ' y_c ' a cada valor calculado de ' y '.

Aunque se requiere una multiplicación en el cálculo de cada parámetro, el multiplicador es una potencia de '2', de modo que la multiplicación puede implantarse como una operación de cambio lógico.

Todas las otras operaciones son simplemente adiciones o sustracciones enteras. El procedimiento que sigue es un código de algoritmo de circunferencia.

La entrada para el procedimiento son las coordenadas del centro y del radio de la circunferencia. El procedimiento, carga el arreglo del buffer de estructura con puntos situados en la circunferencia por medio de llamadas de operaciones `set_pixel`.

El algoritmo es el siguiente :

procedure bres_circle (x_center, y_center, radius : *integer*):

var

p, x, y : *integer*;

procedure plot_circle_points;

begin

set_pixel (x_center + x, y_center + y);

set_pixel (x_center - x, y_center + y);

set_pixel (x_center + x, y_center - y);

set_pixel (x_center - x, y_center - y);

set_pixel (x_center + y, y_center + x);

set_pixel (x_center - y, y_center + x);

set_pixel (x_center + y, y_center - x);

set_pixel (x_center - y, y_center - x);

end; {plot_circle_points}

begin {bres_circle}

x := 0,

y := radius

p := 3 - 2 * radius;

while x < y **do begin**

plot_circle_points;

if p < 0 **then** p := p + 4 * x + 6

else begin

p := p + 4 * (x - y) + 10;

y := y - 1

end; {if p not < 0}

x := x + 1

end; {while x, < y}

if x = y **then** plot_circle_points

end; {bres_circle}

Etapas para calcular las coordenadas enteras más próximas a una circunferencia definida anteriormente :

1. Seleccionar la primera posición para el despliegue como :

$$(x_1, y_1) = (0, r)$$

2. Calcular el primer parámetro como :

$$p_1 = 3 - 2r$$

Si ' $p_1 < 0$ ', la siguiente posición es $(x_1 + 1, y_1 - 1)$. De lo contrario, la siguiente posición es $(x_1 + 1, y_1 - 1)$.

3. Continuar incrementando la coordenada 'x' en pasos unitarios y calcular cada parámetro sucesivo 'p' a partir del anterior. Si para el parámetro anterior se encontró que ' $p_i < 0$ ', entonces :

$$p_{i+1} = p_i + 4x_i + 6$$

En caso contrario (para ' $p_i \geq 0$ ') :

$$p_{i+1} = p_i + 4(x_i - y_i) + 10$$

Por lo tanto, si ' $p_i + 1 < 0$ ', el siguiente punto seleccionado es $(x_i + 2, y_i + 1 - 1)$. La coordenada 'y' es ' $y_i + 1 = y_i$ ', si ' $p_i < 0$ ' o bien ' $y_i + 1 = y_i - 1$ ', si ' $p_i \geq 0$ '.

Repetir los procedimientos del paso 3 hasta que las coordenadas 'x' y 'y' sean iguales.

III.5.3 ELIPSES

Un algoritmo de trazo de circunferencias puede ampliarse para dibujar circunferencias o elipses. En la Figura III-19 se muestra una orientación de una elipse, con ' r_1 ' que marca el eje semimayor y ' r_2 ' como el eje semimenor. La forma estándar de la ecuación elíptica es :

$$\left(\frac{x - xc}{r_1}\right)^2 + \left(\frac{y - yc}{r_2}\right)^2 = 1 \quad 3-24$$

Mediante el uso de las coordenadas polares ' r ' y ' θ ', también se pueden escribir las ecuaciones elípticas en forma paramétrica :

$$\begin{aligned} x &= xc + r_1 \cdot \cos \theta \\ y &= yc + r_2 \cdot \sin \theta \end{aligned} \quad 3-25$$

El algoritmo de Bresenham puede modificarse para generar formas elípticas utilizando la ecuación '3-24', en vez de la ecuación de la circunferencia, en la evaluación del parámetro ' p '. Es decir, para una elipse con centro en el origen, se pueden expresar los valores de ' y ' en la forma :

$$y^2 = r_2^2 \left(1 - \frac{x^2}{r_1^2}\right) \quad 3-26$$

La única diferencia en el algoritmo está en la forma de los parámetros ' p '. Una elipse se traza en posición arbitraria agregando desplazamientos a los valores de ' x ' y ' y ' de salida, como en la generación de posiciones de la circunferencia.

Para ofrecer a los usuarios la capacidad de generar elipses (y circunferencias), un paquete de gráficas podría incluir un comando de la forma :

ellipse (xc, yc, r1, r2)

Las coordenadas del centro de la elipse se asignan a los parámetros ' x_c ' y ' y_c ' y los ejes semimayor y semiminor se especifican en ' r_1 ' y ' r_2 '. Si los valores asignados a ' r_1 ' y ' r_2 ' son iguales, el sistema despliega una circunferencia. Estos parámetros se pasan al algoritmo de generación de elipses el cual traza la figura especificada.

Los comandos para generar una circunferencia y elipses a menudo incluyen la capacidad de trazar secciones curvas especificando parámetros de los extremos de la línea. La extensión de la lista de parámetros permiten la especificación de los valores angulares iniciales y finales de un arco, como se ilustra en la Figura III-20.

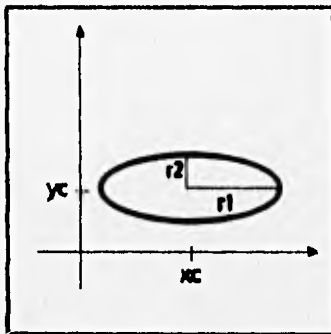


FIGURA III-19
Elipse centrada en (x_c, y_c) con eje semimayor ' r_1 ' y eje semiminor ' r_2 '

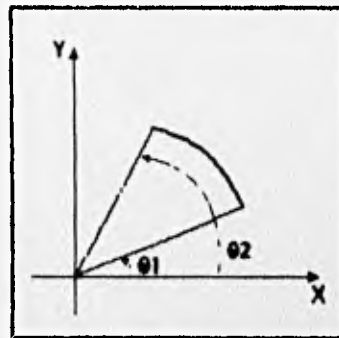


FIGURA III-20
Arco circular especificado por los ángulos inicial y final. El centro de la circunferencia está en el origen coordenado

III.5.4 OTRAS CURVAS

Los procedimientos para desplegar varias curvas utilizan métodos similares a aquellos que generan circunferencias y elipses. Las curvas se encuentran comúnmente incluyendo **funciones seno, funciones exponenciales, polinomiales, distribución de probabilidad y funciones de spline.**

Si se puede expresar una curva en forma funcional, $y = f(x)$, los valores de 'y' pueden calcularse y trazarse con respecto a un intervalo especificado de 'x'. Los valores coordenados deben redondearse al entero más próximo y la trayectoria de la curva puede llenarse con puntos individuales o segmentos rectilíneos. En muchas aplicaciones, las curvas aproximadas con segmentos de recta son más adecuadas. Un método de trazo de punto deja brechas en la curva en áreas donde la magnitud de la pendiente es mayor que '1'. Evitar las brechas con un método de trazo de punto significa que se debe obtener la función inversa, $x = f^{-1}(y)$ y calcular los valores de 'x' para valores dados de 'y' siempre que la magnitud de la pendiente se vuelva grande.

Las consideraciones de simetría mejoran la eficiencia de algunos algoritmos de generación de curvas. Muchas curvas tienen patrones repetidos, de manera que puede ser posible obtener más de un punto sobre la curva con un sólo cálculo.

Para una curva definida por un conjunto de datos de puntos coordenados discretos, se debe graficar la curva de otra forma. Un método consiste simplemente en trazar los puntos de datos individuales y conectarlos con segmentos rectilíneos. Otro punto de vista consiste en utilizar técnicas de ajuste de curvas para aplicaciones de diseño.

III.6 GENERACION DE CARACTERES

Los modelos de retícula rectangular se utilizan comúnmente para definir y trazar caracteres. La Figura III-21 ilustra un modelo de bits de la letra 'B', definida en una matriz de puntos de 8 x 8 para usarse con un sistema rastreador de dos niveles. Cuando este modelo se copia en algún área del buffer de estructura, los bits '1' designan qué posiciones de pixeles se desplegarán en el monitor. Las retículas rectangulares para definiciones de caracteres varían de cerca de 5 x 7 a 9 x 14 o más y se utilizan con sistemas rastreadores y vectoriales, aunque algunos sistemas vectoriales generan caracteres con segmentos de líneas.

Los modelos de caracteres estándar de letras, números y otros símbolos son predefinidos y se almacenan en memoria solo para lectura (ROM). En algunos sistemas, pueden acomodarse modelos de caracteres definidos por el usuario adicionales, permitiendo fuentes especializadas.

Además de permitir a los usuarios definir caracteres especiales, los paquetes de gráficas ofrecen opciones de varios tipos de manipulación de caracteres. Los caracteres pueden cambiarse con relación entre sí para ofrecer efectos especiales o espaciamiento, y los modelos de caracteres pueden rotarse o escalarse para variar su tamaño.

Los comandos del usuario para producir cadenas de caracteres se ofrecen con todos los paquetes de gráficas. Un comando de carácter básico puede definirse como:

text (x, y, string)

Al parámetro **string** se le asigna cualquier secuencia de caracteres, la cual se despliega comenzando en la posición de texto (x, y). Por ejemplo, la proposición:

text (100, 450, "population distribution")

podría utilizarse como un rotulo en una gráfica de distribución.

La posición del texto (x, y) fija la localidad coordinada de la esquina inferior izquierda del primer carácter de la cadena horizontal que se desplegará. Esto ofrece la referencia para copiar las definiciones de la retícula de caracteres en el buffer de estructura. El comando **text** es implantado por una rutina que coloca los modelos de bits de los caracteres en el arreglo rastreador, uno a la vez de izquierda a derecha comenzando en la posición del texto. Los paquetes de gráficas a menudo permiten otras orientaciones de la cadena, como la escritura vertical. Con esta opción, la posición del texto (x, y) puede interpretarse en forma diferente por parte de la rutina **text**.

Otro comando de carácter adecuado es aquel que coloca un carácter designado (símbolo marcador) en una o más posiciones seleccionadas. Este comando puede definirse en forma análoga al comando de línea :

polymarker (n, x, y)

Los parámetros de 'n', 'x' y 'y' en **polymarker** tienen el mismo significado como el comando **polyline**. La diferencia es que **polymarker** ocasiona que se coloque un carácter predefinido en cada una de las 'n' posiciones coordinadas especificadas en los arreglos 'x' y 'y'. El tipo de símbolo que se utiliza en **polymarker** depende de la implantación que se use, pero por ahora se supone que se empleará un asterisco. La Figura III-22 ilustra un trazo de un conjunto de datos con la proposición :

polymarker (6, x, y)

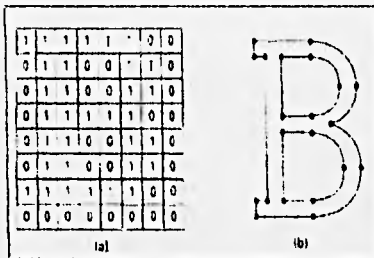


FIGURA III-21
Posible modelo de bits de la letra B, mediante el uso de una retícula rectangular de 8 x 8

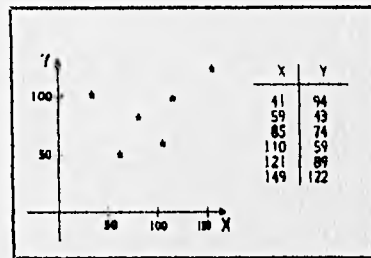


FIGURA III-22
Secuencia de puntos de datos especificada con el comando **polymarker**

La implantación de **polymarker** es llevada a cabo por una rutina que copia repetidamente la definición de la retícula marcada en el buffer de estructura en la posición coordinada designada. Como sucede con el comando **text**, las coordenadas especificadas fijan la posición de la esquina inferior izquierda de la retícula de caracteres.

III.7 CONJUNTOS DE INSTRUCCIONES PARA PROCESADORES DE DESPLIEGUE

Las funciones primitivas de salida que se expresan en un programa de aplicación son convertidas por el sistema de gráficas en una forma adecuada para que el procesador de despliegue genere la imagen en un dispositivo de salida. El conjunto de instrucciones específicas que utiliza el procesador de despliegues para generar la salida depende del tipo de dispositivo que se emplea.

III.7.1 SISTEMAS DE RASTREO CON RASTREADO

Varios registros se ponen a disposición de los procesadores en un **sistema rastreador** a fin de almacenar posiciones coordinadas y diversas instrucciones. Pueden utilizarse cuatro registros para contener valores coordinados de extremo de líneas, parámetros de circunferencias, elipses y posiciones de cadenas de caracteres o marcadores. El tamaño de estos registros se determina por la resolución del sistemas. Si el rastreador está diseñado para desplegar 1024 x 1024 pixeles, los registros coordinados necesitan 10 bits para almacenar los valores coordinados de 0 a 1023. Otros registros se emplean para almacenar operaciones de salida y las instrucciones para procesar los comandos primitivos de salida. El tamaño común de los registros de instrucciones varía de 12 a 24 bits.

Los formatos de instrucciones del procesador de despliegues se organizan en campos de código 'op' y de dirección. El campo de código 'op' determina el tipo de operación que se realizará, como la carga de un registro, el trazo de una línea o el despliegue de un carácter. Los campos de dirección se utilizan para especificar localidades de registros o de memoria.

Cada comando primitivo de salida de un programa de aplicación se compila en un formato de instrucciones correspondiente. La rutina de conservación de rastreo, carga después la instrucción en un registro y la procesa utilizando un procesamiento implantado en hardware adecuado, como el algoritmo de trazo de líneas de Bresenham o la recuperación de un modelo de trazo

de carácter, estos registros son utilizados para almacenar las coordenadas de un pixel. Inicialmente, el registro 'x' se hace igual a '0' y el registro 'y' se hace igual al valor coordenado de la parte superior de la pantalla (por decir algo, 1023). El valor almacenado en el rastreador de esta posición del pixel se utiliza después para ajustar la intensidad del haz de electrones de un CRT de renovación. Después el registro 'x' se incrementa en '1' y el proceso se repite para el siguiente pixel de la línea de rastreo de la parte superior.

Este procedimiento se repite con cada pixel a lo largo de la línea de rastreo. Luego que se ha procesado el último pixel de la línea de rastreo de la parte superior, el registro 'x' se vuelve a fijar en '0' y el registro 'y' se reduce a '1'. Los pixeles situados en esta línea de rastreo se procesan en turno y el procedimiento se repite por cada línea de rastreo.

III.7.2 SISTEMAS DE RASTREO AL AZAR

Los formatos de instrucciones de los **sistemas vectoriales** son semejantes a los de los sistemas rastreadores. La diferencia principal es que se crea un archivo de despliegue de instrucciones en el proceso de renovación en vez de cargar un buffer de cuadros. Para efectuar el proceso de renovación, la primera instrucción del archivo de despliegue se carga en un registro.

Después se hace referencia de subrutinas adecuadas para desplegar una línea o cadena de caracteres en dispositivos de salida, ya sea utilizando técnicas de generación de vectores analógicos o bien digitales. Después, se incrementa un contador de instrucciones y la siguiente instrucción se recoge del archivo de despliegue y se procesa.

Una vez procesadas todas las instrucciones del archivo de despliegue, el contador de instrucciones se borra y el procedimiento de renovación se repite desde el inicio del archivo. Según el tipo de generador de vectores que se utilice, una línea puede ser renovada por un sistema de rastreo al azar a velocidades que varían desde unos cuantos microsegundos hasta una centésima de microsegundo.

CONCLUSIONES

Son muchas las imágenes que pueden crearse mediante procedimientos de graficación de puntos y trazado de rectas. La velocidad a la que se producen figuras gráficas depende de la capacidad del computador y la eficiencia de los algoritmos de trazado y conversión por barrido. En virtud de que la multiplicación y la división de punto flotante pueden requerir hasta 200 veces más tiempo que la suma entera, los procedimientos de graficación deben apoyarse de la aritmética entera siempre que sea posible para obtener mejores resultados en cuanto al tiempo de proceso y de generación de gráficas.

En el siguiente capítulo se muestran algunas de las aplicaciones que tienen estas técnicas de graficación.

CAPITULO IV

TRANSFORMACIONES GEOMETRICAS BIDIMENSIONALES Y TRIDIMENSIONALES

- **DEFINICION DE
GRAFICAS**
- **TIPOS DE
TRANSFORMACIONES**
- **VISTAS
BIDIMENSIONALES Y
TRIDIMENSIONALES**
- **TECNICAS DE
DESPLIEGUE**



TRANSFORMACIONES GEOMETRICAS BIDIMENSIONALES Y TRIDIMENSIONALES

Con los procedimientos vistos en el capítulo anterior para el despliegue de primitivas de salida y sus atributos, se pueden crear infinidad de formas de figuras y gráficas. En muchas aplicaciones existe la necesidad de alterar o manipular despliegues necesitando algunas veces reducir o aumentar el tamaño de un objeto o gráfica. También algunas veces se desea probar la apariencia de imágenes reacomodando las posiciones y los tamaños de las partes de la misma en forma relativa. Estas diversas manipulaciones se llevan a cabo aplicando transformaciones geométricas adecuadas. Los principales tipos de transformación son la rotación, la escalación y la traslación; las cuales serán definidas en este capítulo para gráficas bidimensionales y tridimensionales.

IV.1 DEFINICION DE GRAFICAS

Una **gráfica** es una portadora de información que reproduce de forma concreta circunstancias complejas y compactas, también se puede decir que es la representación de datos numéricos por medio de líneas.

La fundamental diferencia entre las **gráficas bidimensionales** y las **gráficas tridimensionales**, es que las **gráficas bidimensionales** son planas mientras que las **gráficas tridimensionales** tienen volumen. La Figura IV-1 ilustra 2 polígonos planos en espacio de 2-D y dos simples polígonos estructurados en un espacio de 3-D. Para tener una visión clara, las líneas del frente de los

objetos tridimensionales son gruesas. Es necesario notar que las gráficas tridimensionales son creadas a partir de gráficas primitivas bidimensionales.

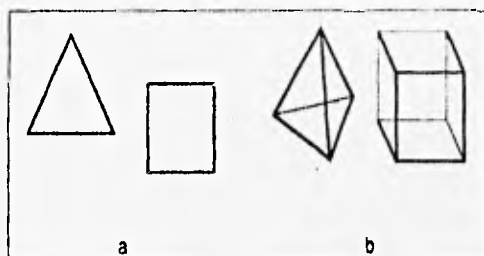


FIGURA IV-1
(a) Triángulo y cuadrado bidimensional (b) Pirámide y cubo tridimensional

IV.1.1 GRAFICAS BIDIMENSIONALES

Un dibujo de **gráficas bidimensionales** está descrito por dos ejes, una eje corriendo horizontalmente, llamada eje 'x', y una eje corriendo verticalmente llamada eje 'y'. Estas ejes forman un sistema coordinado, donde el origen es (0, 0).

Las líneas dibujadas en la cuadrícula del plano gráfico (rastreador) son llamadas **vectores**. Para dibujar un vector, nosotros podemos especificar 2 puntos en la pantalla y pedir a la computadora que dibuje una línea entre ellos. La Figura IV-2 ilustra vectores dibujados a partir de las coordenadas (2, 3) a la (2, 8) y desde la (4, 5) hasta la (9, 5).

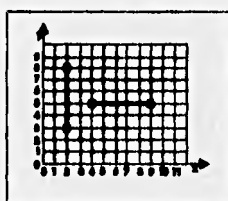


FIGURA IV-2
Vectores dibujados de (2, 3) a (2, 8) y de (4, 5) a (9, 5)

Similarmente para dibujar un rectángulo, nosotros podemos especificar sus 4 esquinas o vértices. Por ejemplo, existen aplicaciones que contienen herramientas para dibujar rectángulos, facilitando estas tareas al usuario, ya que solo necesita seleccionar la herramienta con un click del mouse y luego indicar el punto de partida y arrastrar el puntero diagonalmente hacia el punto final, definiendo así los parámetros del rectángulo.

En la Figura IV-3 se ilustra como se dibuja un rectángulo.

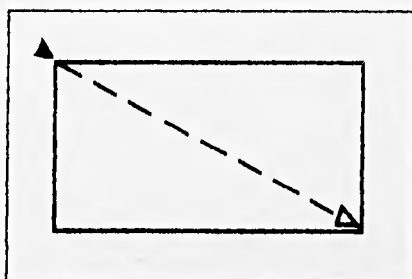


FIGURA IV-3
Dibujando un rectángulo con una aplicación

En la Figura IV-4 se ilustra el código del software que un programador puede utilizar para dibujar el rectángulo.

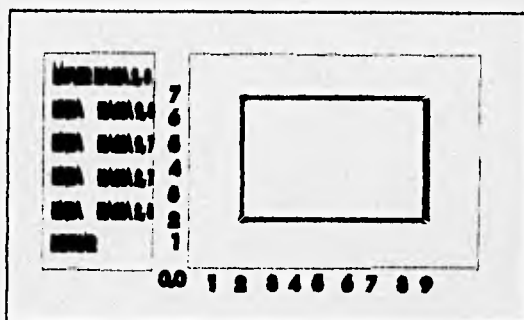


FIGURA IV-4
Código del rectángulo (2, 3)
(9, 3) (9, 7) (2, 7)

IV.1.2 GRAFICAS TRIDIMENSIONALES

Se denomina **gráfica tridimensional** a la imagen analítica de objetos espaciales, descritos de forma numérica o matemática, en un plano bidimensional.

Un **sistema tridimensional** de programas gráficos tiene la misión de crear una estructura adecuada de datos sobre los objetos a ser dibujados, sobre la manipulación de los datos y sobre su emisión gráfica. Una cadena de programas de este tipo convierte al ordenador en una cámara, con la limitación de un almacenamiento de abstracción del objeto imaginario. Con respecto al criterio del dibujo, a la dirección de la imagen y a la relación de la figura no existen restricciones, debido a la posibilidad de poder efectuar cualquier transformación geométrica.

Las **gráficas tridimensionales** son mucho más complejas que las gráficas bidimensionales por muchas razones. Primero, la computadora debe de realizar más cálculos para crear dibujos tridimensionales, ya que estos deben de ser proyectados sobre superficies bidimensionales. Adicionalmente, con gráficas tridimensionales, tenemos la libertad para dibujar en un espacio voluminoso. Un dibujo en 3D puede tener muchos lados y puede moverse en un espacio hacia adelante o hacia atrás, lo que no es posible con un dibujo bidimensional.

Para entender el espacio tridimensional, hay que pensar como los ejes 'x' y 'y' forman una superficie plana, tipo una caja rectangular, y que el eje 'z' está conectado por una **esquina** y corre perpendicular a este plano, representando la profundidad de la caja. Un espacio tridimensional consta de 3 ejes, llamado **volumen** que puede ser considerado un espacio infinito en todas direcciones. La Figura IV-5 ilustra el punto de referencia visual posicionado a lo largo del eje 'x'.

Podemos clasificar las aplicaciones de las **gráficas tridimensionales** conforme a si se trata para una representación de objetos existentes ('reales') o bien, para un diseño de nuevas figuras. Por ejemplo, un objeto sólido podría describirse como una estructura de líneas o bien, como un conjunto de superficies planas (Figura IV-6), en otra aplicación, tendríamos que especificar una representación que incluya líneas y superficies curvas. En las aplicaciones de diseño con ayuda de la computadora, el objetivo consiste en crear objetos construyendo y manipulando modelos

para formar nuevas figuras tridimensionales. Por ejemplo, las carrocerías y cuerpos de automóviles y aviones se diseñan reacomodando patrones de superficies hasta que se cumplan ciertos criterios de diseño.

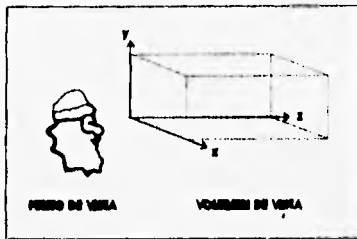


FIGURA IV-5
El punto de referencia visual y la vista tridimensional creada por los ejes 'x', 'y' y 'z'



FIGURA IV-6
Un objeto 3D puede representarse como una estructura de líneas o bien como un conjunto de superficies planas

EL PUNTO DE REFERENCIA VISUAL EN EL ESPACIO TRIDIMENSIONAL

A partir de la descripción anterior, el volumen puede ser considerado un espacio tridimensional donde el usuario puede ver dentro de él. Las librerías de gráficas tridimensionales de los paquetes gráficos, habilitan al ojo del observador para ver el volumen de cualquier objeto desde cualquier ángulo. Esta capacidad, es una analogía a una cámara (llamada cámara sintética), la perspectiva de la escena que es desplegada desde el CRT es como una toma instantánea desde cualquier ángulo. El sistema coordinado de una cámara sintética es semejante al sistema coordinado 'x', 'y', 'z', que es un espacio definido matemáticamente y que crea una ilusión tridimensional, ya que el sistema gráfico distingue la geometría tridimensional de la escena. Un sistema gráfico es capaz de producir diversas imágenes de los objetos desde cualquier ángulo.

En la Figura IV-7 (a) se muestra un punto de referencia visual (o cámara) enfocada en una imagen es un espacio tridimensional, y en la Figura VII (b) se muestra el resultado cuando el

punto de referencia visual es fijado en esta localidad. La ventaja de esta aproximación es que el objeto en esta perspectiva puede ser calculado y colocado una vez en la localidad deseada con respecto a otra.

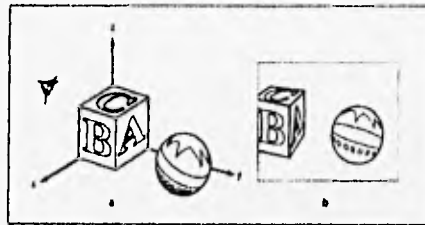


FIGURA IV-7
(a) El punto de referencia visual en espacio tridimensional (b) Lo que el usuario percibe

SIMULACION TRIDIMENSIONAL

La **simulación** de objetos geométricos tridimensionales utilizando formas bidimensionales, es un efecto llamado **dos-y-media-dimensiones** (two-and-a-half-dimension), este es utilizado en gráficas de animación secuencial. La Figura IV-8, ilustra una escena donde se aplica el efecto de **dos-y-media-dimensiones**. La gráfica contiene varios cilindros en diferentes orientaciones, hay que notar la apariencia tridimensional de un cilindro utilizando formas bidimensionales, tales como un círculo y un rectángulo.

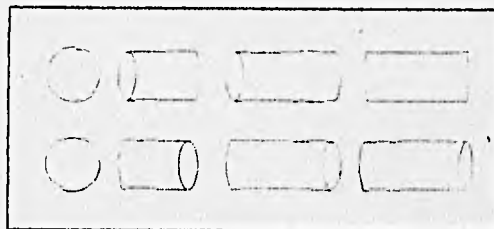


FIGURA IV-8
Un gráfica de dos-y-media-dimensiones

Estos objetos no pueden ser girados o rotados en un espacio tridimensional ya que son creados a partir de formas bidimensionales para crear sólo una ilusión tridimensional. Así nosotros estamos restringidas a ver sólo la vista frontal de la perspectiva y no podemos ver los lados o el fondo del objeto. La Figura IV-9 ilustra una serie de cilindros y de polígonos que simulan una apariencia de tercera dimensión.

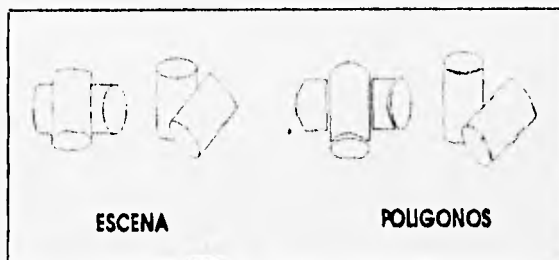


FIGURA IV-9
Gráficas bidimensionales utilizadas para simular polígonos tridimensionales

En adición al dibujo de formas bidimensionales para la creación de apariencias tridimensionales, el usuario debe de ser capaz de simular **OBJECT OCCLUSION** (que es el efecto de un objeto bloqueando a otro por encima de éste para simular un espacio tridimensional desde un punto de referencia visual). Para lograr esto, se requiere de un trabajo intensivo por parte del usuario, ya que tiene que dibujar figuras con el efecto de **dos-y-media-dimensiones**.

IV.1.3 REDUCCION DE GRAFICAS DE TERCERA A SEGUNDA DIMENSION

En las gráficas de tercera dimensión, la computadora calcula la geometría tridimensional de los objetos, una imagen es descrita en un espacio tridimensional en términos de su altura, ancho y profundidad, ya que la superficie de la pantalla es plana, el resultado final es entonces una imagen bidimensional: Un objeto tridimensional es reducido a 2 dimensiones antes de ser desplegado. Esto es parte de una secuencia de pasos que ocurren cada vez que un objeto es desplegado o manipulado en alguna forma, y después red desplegado.

Por ejemplo, cuando el usuario rota un objeto hacia una nueva orientación o lo escala a un nuevo tamaño, la computadora calcula las nuevas localizaciones y orientaciones de las gráficas primitivas del dibujo en espacio tridimensional y después traslada esta información a un espacio bidimensional para ser desplegado en la pantalla. El proceso de reducir objetos tridimensionales a objetos bidimensionales para ser desplegados, es llamado **Proyección**.

IV.2 TIPOS DE TRANSFORMACIONES

Las transformaciones geométricas constituyen un medio con el que se pueden construir o modificar imágenes. Los mismos objetos pueden ser movidos a nuevas posiciones y ángulos, muchas aplicaciones requieren movimientos y cambios para la orientación de objetos en una base regular. Los efectos más realistas pueden ser obtenidos por una combinación de ambos movimientos; el de la cámara y el del objeto en movimiento, como es esencialmente simulado en la vida real. Cuando el espectador se encuentra con la vista fija frente a la computadora, éste no siempre detecta quien está en movimiento, si el objeto o la cámara.

Por instancia, si hay sólo un objeto en la pantalla, tal como un carrusel, rotado en el espacio y no hay otro punto de referencia, es probable que no sea claro para el espectador si el objeto fue rotado, o la cámara estuvo en movimiento alrededor de él. De cualquier modo, si hubo un fondo en la escena, semejante a una cerca o a un árbol, estos podrían seguir en la misma localización si sólo se movió el carrusel, pero podría cambiar la orientación si la vista es movida.

Estas operaciones mueven o reorientan objetos y son llamadas **transformaciones**. Estas operaciones trasladan o cambian las localizaciones de los objetos, los adaptan arriba o abajo, en tamaño, y los rotan a una nueva orientación en el espacio. (Estas operaciones tienen altos requerimientos en computación).

Para mover, rotar o escalar un objeto, el usuario hace una solicitud de la aplicación. Las aplicaciones interactivas con el hardware para hacer estos cambios usan algoritmos de

transformaciones que modifican las coordenadas del objeto a través de la multiplicación o de la adición dependiendo del tipo de transformación solicitada. Los algoritmos de transformación usan matrices de transformación que condensan las matemáticas de las operaciones dentro de una forma compacta, habilitando la aplicación para hacer los cambios a las coordenadas del objeto eficientemente.

Si el sistema de gráficas incluye un comando para cada transformación, el usuario típicamente puede mover, rotar o escalar una imagen turnando cada uno de los comandos. La aplicación puede implementar un uso diferente para varios comandos por lo que cada comando ejecuta una de las transformaciones básicas.

IV.2.1 TRASLACION

Traslación de un objeto significa moverlo de una posición a otra nueva posición en la pantalla, esto es conocido como **desplazamiento** que es un efecto algo similar al movimiento de una cámara de video a través de una escena. Una traslación computarizada involucra una matriz que representa la **traslación offset**, la cual es la cantidad de espacio en cada vértice donde el objeto debe ser movido desde la posición actual y trasladado por esa cantidad en cada **axis**. Por instancia, si el usuario mueve un objeto de 2-D, y la traslación offset es (4, 2), cada coordenada en el objeto debe ser trasladada por 4 unidades a la derecha y 2 unidades hacia arriba. La Figura IV-10 ilustra un cuadro 2-D trasladado por (4, 2).

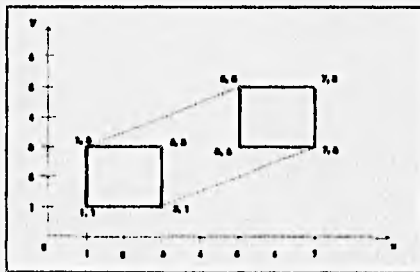


FIGURA IV-10
Una traslación 2D con un offset de (4, 2)

En tercera dimensión, el concepto es similar, excepto que en este habrá además un offset para la coordenada 'z'. La Figura IV-11 ilustra un objeto de 3-D trasladado por un offset de (2, 3, 4). Esto significa que cada punto de las coordenadas del objeto es movido a la derecha por 2 coordenadas en la axisa 'x', hacia arriba por 3 coordenadas en la axisa 'y' y hacia adelante por 4 coordenadas en la axisa 'z'.

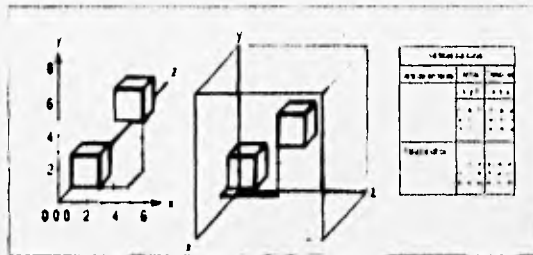


FIGURA IV-11
La Traslación de un cubo antes y después con un offset de (2, 3, 4)

En esta Figura, dos vistas son mostradas: la vista frontal y la vista lateral. La vista frontal ilustra el espacio tridimensional como es ilustrado en otros textos gráficos. La vista lateral provee una segunda perspectiva, para ilustrarla, y para la axisa 'z' está perpendicular al plano creado por (x, y).

IV.2.2 ROTACION

Las **rotaciones** se toman en forma muy diferente en 2-D y 3-D. Una rotación en 2-D es conceptualizada como un pivote que como una rotación, porque las **rotaciones bidimensionales** ocurren alrededor de un simple punto en el plano (x, y). La Figura IV-12 ilustra una rotación con una analogía semejante a las manecillas de un reloj. Las cuales pueden ser rotadas en el sentido de las manecillas de un reloj o en sentido contrario alrededor de un punto. En este caso el punto de rotación es el origen de (x, y).

En tercera dimensión, las rotaciones no ocurren alrededor de un simple punto, pero sí alrededor de una axis especificada. La Figura IV-13 muestra un cilindro rotando en cada una de las tres axis. Para rotaciones tridimensionales, la aplicación pasa al hardware por la solicitud del usuario, la aplicación entonces usa una matriz de rotación para computar el ángulo de rotación alrededor de un punto o de una axis.

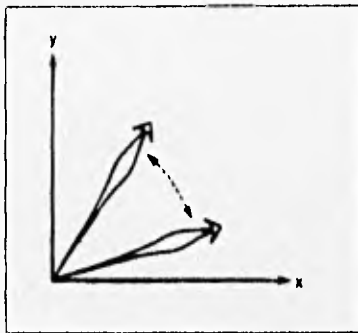


FIGURA IV-12
Una rotación bidimensional

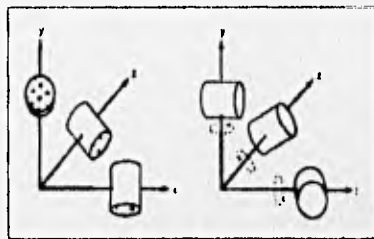


FIGURA IV-13
Rotaciones tridimensional de un cilindro en el eje 'x', 'y' y 'z'

IV.2.3 ESCALACION

La **escalación** produce un efecto que es similar a el acercamiento o alejamiento de un lente de cámara, incrementando o decrementando el tamaño de los objetos, y por esta razón la escalación es algunas veces llamada **zooming**. La escalación de un objeto a un nuevo tamaño implica una multiplicación de matrices que representa el factor de escala por el cual un objeto debe ser incrementado o decrementado. Este factor es multiplicado por las coordenadas de cada vértice en el objeto. Si las proporciones permanecen igual, el factor de escala tiene que ser el mismo para 'x' y 'y' en 2-D, o para 'x', 'y' y 'z' en 3-D. Un número menor que 1 decrementa la escala, mientras que un número mayor que 1 incrementa la escala. La Figura IV-14 ilustra un objeto bidimensional y un objeto tridimensional, cada escala esta dada por un factor de 3. Además es

necesario notar que ese escalafón traslada un objeto a una nueva posición, porque el objeto no está centrado en la coordenada (0, 0).

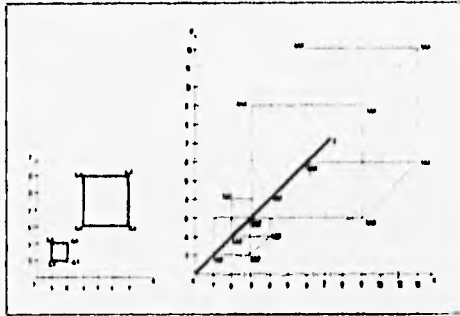
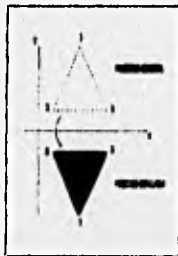


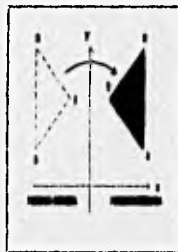
FIGURA IV-14
Un objeto 2D y 3D incrementados en escala con un factor de 3

IV.2.4 REFLEXION

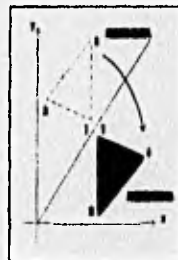
Una **reflexión** es una transformación que produce una imagen de espejo de un objeto. La imagen del espejo se genera relativa a un eje de reflexión. En una **reflexión con respecto al eje 'y'** las coordenadas 'x' son modificadas, mientras mantiene inalteradas las coordenadas 'y'. Otro tipo de reflexión es aquella que modifica las coordenadas 'x' y 'y', por ser una reflexión **relativa al origen coordenado**. La Figura IV-15 ilustra los diferentes tipos de transformaciones con respecto a cada uno de los ejes.



Reflexión de un objeto con respecto al eje 'x'



Reflexión de un objeto con respecto a la recta 'y = x'



Reflexión de un objeto con respecto al eje 'y'

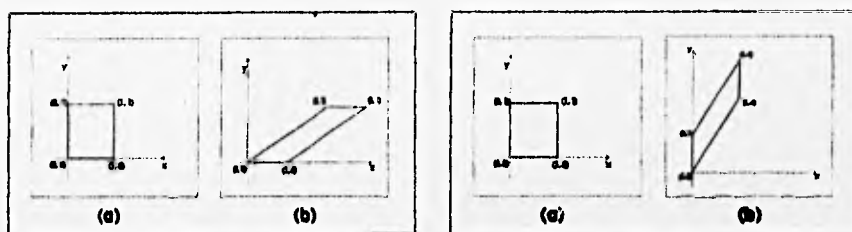


Reflexión de un objeto relativa al origen coordenado

FIGURA IV-15

IV.2.5 CORTE

Este tipo de transformación produce distorsiones de figuras que representan un efecto de doblez o de corte, como si un objeto estuviese compuesto por capas que se hacen deslizar la una sobre la otra. Dos transformaciones de corte comunes son aquellas del corte en la dirección 'x' y el corte en la dirección 'y'. La Figura IV-16 muestra los diferentes tipos de corte con respecto al eje 'x' y al eje 'y'.



Cuadrado unitario convertido en un paralelogramo (utilizando un corte en la dirección 'x') Cuadrado unitario convertido en un paralelogramo (utilizando un corte en la dirección 'y')

FIGURA IV-16

IV.3 VISTAS DE BIDIMENSIONALES Y TRIDIMENSIONALES

IV.3.1 VISTAS BIDIMENSIONALES

Cuando los objetos son dibujados, modificados y desplegados, una serie de eventos toma ese lugar transformando los datos de la imagen "atrás de las escenas", estos cambios ocurren rápidamente e imperceptiblemente. Estos eventos son llamados **vista de tubería**, las diferentes

etapas en la vista de tubería incluyen la separación de coordenadas de sistemas para el usuario, el software y el hardware, ya que definen la orientación del objeto u objetos en el espacio. Los cálculos requeridos por los pasos en la vista de tubería son intensos, particularmente si la vista o la posición de los objetos en la escena cambian continuamente.

COORDENADAS DE MODELO

En los sistemas gráficos, las comunicaciones ocurrirán entre el usuario, las librerías gráficas y el hardware gráfico. Parte del trabajo de software es proveer al usuario un lugar de trabajo lógico, y para realizar la entrada de datos, los cuales no deben ser desplegados en píxeles, ya que podría ser difícil para los usuarios gráficos describir un objeto en unidades de píxel.

Si las aplicaciones describen objetos directamente en píxeles, estos objetos no podrían trabajar consistentemente en dispositivos hardware. Por estas razones, los sistemas de coordenadas de modelos y los sistemas de coordenadas mundiales han sido ideados por el usuario.

Los sistemas de coordenadas de modelo de sistemas de coordenadas son aplicables a un simple objeto gráfico. Por instancia, si el objeto es el modelo de un juguete, probablemente es lógico para el uso de sistemas de coordenadas de modelo, definirlos por pulgadas o centímetros. Cada objeto en una escena es definido por su propio sistema de coordenadas de modelo, los cuales significan que cada objeto potencialmente tiene un sistema de medidas únicas.

COORDENADAS MUNDIALES

Los sistemas de coordenadas mundiales, o mundo de coordenadas, es el espacio donde existe la separación de sistemas de coordenadas de modelo para una imagen en donde son combinados en uno. Todos los objetos en una escena son definidos y trazados en coordenadas mundiales. Los sistemas de medidas en el mundo coordinado dependen de cada aplicación en particular.

Algunas porciones de los datos en el mundo coordinado, son posiciones en una ventana, ya que esta región define el área de interés dentro del sistema del mundo coordinado. En la

Figura IV-17, la separación de las coordenadas de modelo del tren están trazados dentro de una ventana en el mundo coordinado.

METODO CLIPPING O DE RECORTE

Siguiendo la imagen para el mundo coordinado, la ventana determina que porción de la gráfica va a ser desplegada. Algunas partes de la figura quedan fuera del mundo coordinado, por lo que la ventana descarta que parte de la figura va a ser recortada, esto se lleva a cabo por medio de un algoritmo llamado **método clipping** o de recorte. Este método libera la librería de gráficas del trabajo de primitivas dibujadas que quedan fuera de las fronteras de la ventana, y remueve algunas partes distraídas de la escena para ayudar a que el espectador atienda la porción proyectada.

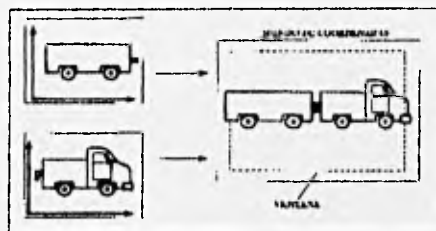


FIGURA IV-17
Coordenadas de modelo para el
tren trazado en el mundo
coordinado

Clipping es un proceso de tiempo consumido por el sistema de gráficas, porque requiere de una serie de exámenes. Estos exámenes determinan que primitiva está dentro de la ventana (no requieren cortado), **parcialmente dentro** de la ventana (las primitivas son cortadas y las porciones visibles desplegadas), o **totalmente fuera de la ventana**. Los algoritmos de recorte tienen que ser eficientes ya que en un simple dibujo pueden estar miles de primitivas para ser consideradas por un posible cortado, y si el objeto o la vista está moviéndose, el **cortado es un proceso on going**. Existen dos funciones que requieren de menos cálculos y son nombradas apropiadamente: cuando las primitivas están en su totalidad fuera de la ventana, el proceso es llamado **rechazo trivial**, cuando están en su totalidad dentro de la ventana, es llamado **aceptación trivial**.

DISPOSITIVOS COORDENADOS.

Estos sistemas coordenados, definen la rejilla como un espacio de dos dimensiones descrito por las ejes 'x' y 'y'. Los sistemas de medidas describen en pixeles a los parámetros de las superficies de despliegue de los dispositivos gráficos. Estos sistemas coordenados son llamados también **dispositivos coordenados** o **dispositivos de espacio** porque ellos son específicos para las gráficas desplegadas en él.

DISPOSITIVOS COORDENADOS NORMALIZADOS

Los dispositivos coordenados y los coordenadas mundiales que son interfaces para otros sistemas coordenados ya que contienen librerías gráficas llamadas **dispositivos de coordenadas gráficos normalizados (NDCs)**. El mundo coordinado es trazado dentro del espacio de NDC, el cual define al vipuerto como una porción de la superficie de despliegue seleccionado como el área visual. La Figura IV-18 ilustra un tren de juguete cortado por la ventana del mundo coordinado y después trazado en la ventana del vipuerto en un espacio NDC.

Después de la ventana del mundo coordinado, se traza al vipuerto NDC. Posteriormente el mismo tren se traza en pixeles a un dispositivo de espacio. Muchas vistas del vipuerto en la superficie pueden ser diseñadas por medio de vistas separadas del mismo objeto, la Figura IV-19 ilustra el tren de juguete en tres vipuertos.

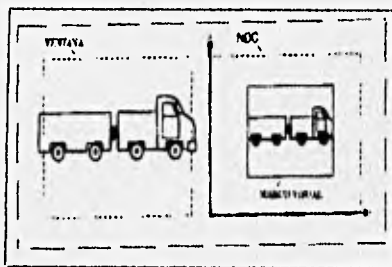


FIGURA IV-18
Trazado de una ventana sobre un vipuerto

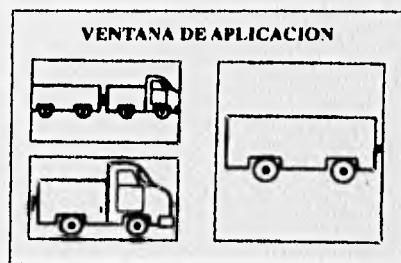


FIGURA IV-19
El tren mostrado en tres vipuertos separados

IV.3.2 VISTA TRIDIMENSIONAL.

Los pasos analizados anteriormente son utilizados tanto en imágenes bidimensionales como en imágenes tridimensionales. Dos fases deben de ser adicionadas al proceso de vistas tridimensionales: uno el de la fase de **cortado** y otra en el tiempo de **despliegue**.

CORTES EN TERCERA DIMENSION

Una imagen de tridimensional debe ser cortada en altura y anchura para convertirse en una imagen bidimensional y con profundidad. El corte en la dimensión de profundidad de volúmenes tridimensionales es llamado **corte-z** porque los algoritmos están relacionados con el espacio a lo largo de la axisa 'z'. Un **corte plano** se usa para definir los límites del área de una imagen. Las librerías utilizadas deben evaluar que partes de los objetos pasan parcialmente o completamente antes de ser dibujada. La Figura IV-20 ilustra un cilindro en un espacio de tridimensional, los cortes de frente y de la parte posterior indican los límites de despliegue.

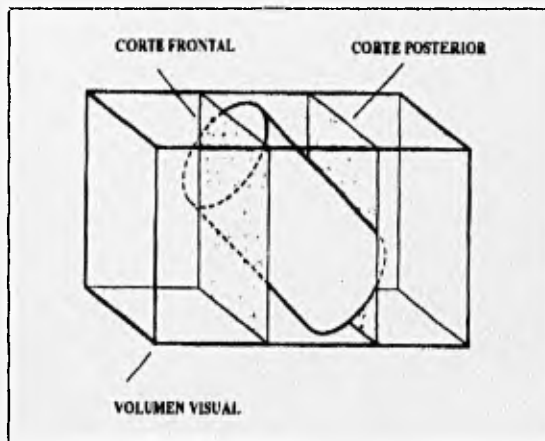


FIGURA IV-20
Una imagen con volumen, limitada por cortes planos frontales y posteriores

IV.4 TÉCNICAS DE DESPLIEGUE

IV.4.1 PROYECCION

Para trazar coordenadas tridimensionales hacia coordenadas bidimensionales en una superficie de despliegue, la imagen tridimensional es proyectada sobre un plano bidimensional. Las **proyecciones** reducen las dimensiones de una imagen bidimensional para ser desplegadas en la pantalla de la computadora. Esto es conceptualizado con rayos llamados **proyectores**, estos proyectores proceden punto por punto de la imagen tridimensional a la superficie bidimensional.

Las **proyecciones** pueden ser ejecutadas en paralelo o en perspectiva. La Figura IV-21 ilustra una imagen tridimensional proyectada en una superficie bidimensional utilizando la **proyección en paralelo**. Como los proyectores son paralelos a la axisa 'z' y con la proyección en perspectiva son conocidos como **proyectores de punto fuga**.

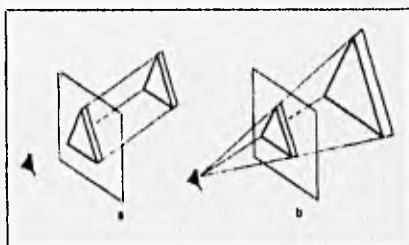


FIGURA IV-21
(a) Proyección paralela
(b) proyección en perspectiva

El corte de profundidad es importante en la proyección de perspectivas, por lo que no hay límite en las dimensiones de volumen. Las librerías de gráficas no pueden dibujar objetos en el trayecto hacia el infinito en el mismo punto, la escala es más pequeña que cuando la imagen es trasladada dentro de un dispositivo de espacio, la librerías por tanto, no puede dibujar más que un grupo de puntos y líneas; este resultado se muestra en la pantalla como un lunar. Limitando la profundidad del espacio libre para dibujar relieves, las librerías intentan dibujar objetos distantes en espacio,

para hacerlos distinguibles al eje. Después del corte y de la proyección de imágenes tridimensionales a un plano bidimensional, la imagen es trazada hacia el vpuerto y después hacia los pixeles en el espacio de la pantalla

La representación de un objeto sólido en una **superficie de visión**, por lo general contiene información de profundidad para que un observador pueda identificar con facilidad cuál es la parte frontal y posterior del objeto. La Figura IV-22 ilustra la ambigüedad que puede originarse cuando un objeto se despliega sin información de profundidad. Existen varias técnicas que pueden utilizarse para incluir información de profundidad en la representación bidimensional de objetos sólidos; la elección de la técnica de despliegue depende de los requisitos de la aplicación.

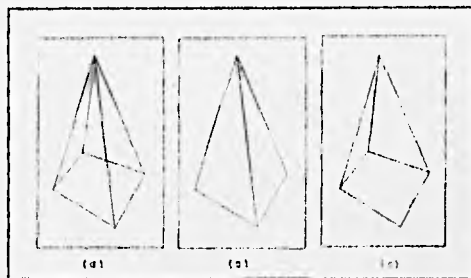


FIGURA 4V-22
 (a) Representación con estructura de alambre de una pirámide; (b) la figura no contiene información de profundidad para indicar si la dirección de visión es desde arriba; (c) la figura no contiene información de profundidad para indicar si la dirección de visión es desde abajo

PROYECCION EN PARALELO

Un objeto sólido puede representarse en dos dimensiones proyectando puntos sobre la superficie del objeto a lo largo de líneas paralelas sobre una superficie plana. Al seleccionar diferente posición de observación, se pueden proyectar puntos visibles sobre la superficie con el fin de obtener diferentes vistas bidimensionales de un objeto, como en la Figura IV-23. En una **proyección en paralelo**, las líneas paralelas sobre la superficie del objeto se proyectan en líneas paralelas sobre el plano de visión bidimensional. Esta técnica se utiliza en ingeniería y en trazos arquitectónicos para representar al objeto con un conjunto de vistas que mantienen proporciones

relativas del mismo. El aspecto del objeto solido puede reconstruirse despues a partir de las vistas principales

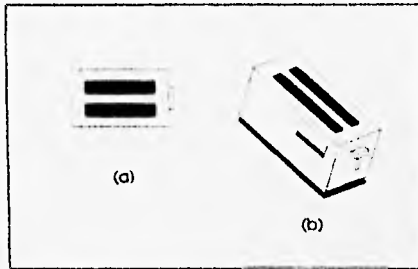


FIGURA IV-23
 Dos vistas con proyección en paralelo de un objeto, que se usan para mostrar proporciones relativas. Las líneas sobre el objeto que son paralelas siguen siéndolo en las vistas de proyección

PROYECCION EN PERSPECTIVA

Otro método para representar la profundidad de una superficie bidimensional es la **proyección en perspectiva**. En vez de proyectar puntos en líneas paralelas, las proyecciones en perspectiva cambian los tamaños de los objetos de modo que aquellos que están más alejados del punto de referencia visual se despliegan de menor tamaño que los que están más próximos a este mismo punto de referencia. Las líneas paralelas sobre la superficie de un objeto se proyectan ahora en líneas que tienden a convergir. Los objetos desplegados con proyecciones en perspectivas parecen más naturales, ya que ésta es la manera en que el ojo y los lentes de una cámara forman imágenes.

IV.4.2 INDICACION DE LA INTENSIDAD

Un método simple para indicar la profundidad en un despliegue, consiste en **variar la intensidad de las líneas** respecto a la distancia del punto de referencia visual. La Figura IV-24 muestra un objeto con líneas frontales destacadas y trazadas más anchas que las posteriores.

Las técnicas de indicación de la intensidad más complejas emplean varias intensidades, de manera que la brillantez decrezca gradualmente para las líneas que están más alejadas del observador.

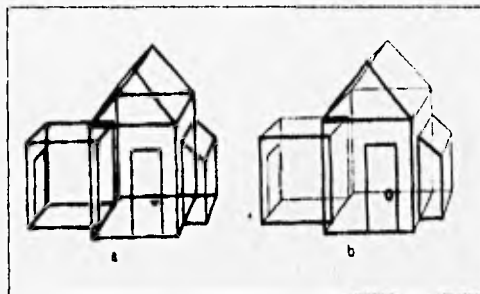


FIGURA IV-24
Realce, forma de ajuste de la intensidad, que puede usarse para marcar las aristas frontales de objetos

IV.4.3 SUPRESION DE LINEAS OCULTAS

Esta técnica es útil cuando los objetos se definen como un conjunto de líneas que representan las aristas de las superficies de los objetos. Para una vista seleccionada, cualquier línea que esté oculta por superficies frontales se suprime antes de que el objeto se despliegue. A pesar de que pueden necesitarse muchos cálculos para localizar todos los segmentos de líneas ocultas en una escena, este método ofrece un despliegue más realista de la información de profundidad en trazos de líneas. En algunas aplicaciones es útil reemplazar las aristas ocultas de objetos con líneas punteadas en vez de suprimirlas todas (Figura IV-25).

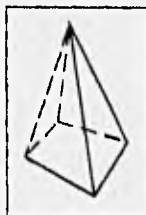


FIGURA IV-25
El despliegue de aristas con líneas punteadas ofrece más información acerca de las superficies anteriores

IV.4.4 SUPRESION Y SOMBREADO DE SUPERFICIES OCULTAS

Cuando los objetos se definen como superficies llenas por modelos de color o sombreado, las **técnicas de supresión de superficies ocultas** se utilizan para hacer visible cualquier superficie anterior que esté oculta por una superficie visible. La supresión de superficies ocultas por lo general, es un proceso complicado y lento, pero ofrece un método altamente realista para desplegar objetos. Se logra mayor realismo, cuando se combina la supresión de superficies ocultas con proyecciones en perspectiva y se incluye sombras y texturas en la superficie.

IV.4.5 VISTAS ESQUEMATICAS Y RECORTADAS

Las técnicas de supresión de líneas y de superficies ocultas pueden combinarse con **vistas esquemáticas y recortadas** para ofrecer más información acerca de la estructura de los objetos tridimensionales. Otra manera de esquematizar un objeto (o descomponer un objeto) en sus partes componentes es la **vista recortada**, la cual suprime parte de las superficies visibles para mostrar la estructura interna.

CONCLUSIONES

Una imagen es creada colocando subimágenes que la componen en posiciones específicas dentro del sistema de coordenadas de mundo; es éste un proceso requerido con una serie de pasos para su montaje. Con frecuencia las subimágenes se crean por separado, las cuales requieren de las transformaciones geométricas; como son la rotación, las escalación, traslación, etc. para posicionar la subimagen correctamente dentro de la imagen. A su vez cualquier parte de la imagen o la imagen completa, puede moverse o distorsionarse aplicando una o varias transformaciones.

El siguiente capítulo mostrará a la Graficación por Computadora como una herramienta para el diseño en Arquitectura.

CAPITULO V

LA GRAFICACION DESDE EL PUNTO DE VISTA ARQUITECTONICO

- **METODOS DE
PRODUCCION PARA
EL DISEÑO EXTERNO**
- **METODOS DE
PRODUCCION PARA
EL DISEÑO INTERNO**



LA GRAFICACION DESDE EL PUNTO DE VISTA ARQUITECTONICO

La mejora y disponibilidad del hardware para la **Graficación por Computadora** y los periféricos utilizados, la han conducido a ser utilizada en la vida de los profesionales y principiantes aficionados en la graficación.

Lo que ha hecho posible esto, es principalmente la facilidad de mejorar el uso de las interfaces gráficas para el usuario, además del poder de cálculo de los CPU'S que se ha incrementado tremendamente en volumen y rapidez para permitir un manejo rápido de las grandes cantidades de datos. Con esto, el campo de la **Graficación por Computadora** ha sobrepasado una valla importante. En el futuro, el hardware para la graficación, los periféricos y el software utilizado continuaran mejorando, y el medio social para la **Graficación por Computadora** crecerá más afinado para brindar más ventajas.

Dentro de las mayores categorías de gráficas por computadora y de las perspectivas de graficación en la arquitectura han llegado a ser un campo establecido y reconocido para un aumento de su uso en oficinas y su aplicabilidad ha llegado a ser bastante alta. A fin de ganar un entendimiento de las gráficas arquitectónicas realizadas por computadora, se ha seleccionado el ejemplo de una casa y sus respectivos interiores donde se aplica el tipo más básico de materiales y se explica el proceso para la entrada de datos además de que se muestra el desarrollo de las perspectivas arquitectónicas de las gráficas por computadora.

V.1 METODOS DE PRODUCCION PARA EL DISEÑO EXTERNO

Los pasos a seguir en el método para el diseño externo son el **Modelado**, la **Interpretación**, el **Mapeo** o **Trazado del diseño**, **Completando la Estructura** y la **Adición de los Detalles**, los cuales explicaremos a continuación.

V.1.1 DATOS DIMENSIONALES (EL MODELADO)

Los datos dimensionales son las dimensiones que van a ser incluidas en el plano del piso, planos verticales y en los terminados. Después de la entrada de todos los datos se creará **la estructura del esqueleto de alambre** donde se tomarán en cuenta las áreas que sean difíciles de observar, tales como las ventanas, los marcos, los barandales del balcón y la parte posterior de la estructura, esto se realiza para estar seguros de que todos los datos sean los correctos, es decir se crea un **modelado**.

V.1.2 DATOS DIMENSIONALES (LA INTERPRETACION)

Para agregar una superficie al **modelo** creado en la etapa de "modelar", usaremos la "interpretación". Con la "interpretación", el trazo se toma dimensionalmente. En este punto, también se añade la textura, más generalmente referida como **materia**, pero en esta parte lo más importante es averiguar que las dimensiones sean las correctas, por lo tanto, el material lo hemos tratado en otro punto.

En realidad, cuando se maneja el material por separado, se acorta grandiosamente el tiempo de calculo en la computadora (llamado **tiempo de interpretar**) y es por lo tanto una opción altamente práctica y eficiente. Cuando las dimensiones coinciden, es también mucho más fácil ver los lugares difíciles, tales como los puntos de conexión entre paredes, etc., por ejemplo, lo

que se necesita checar, es si el bloque de vidrio alrededor de la puerta frontal se conecta con la pared externa, o como se coloca el interior del balcón. Estas dimensiones y relaciones deben verificarse una por una.

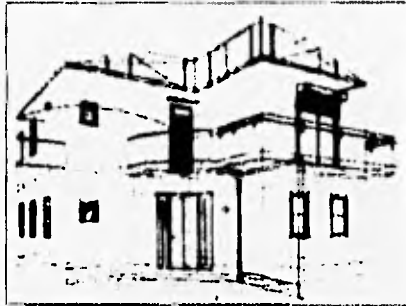
V.1.3 COLOCACION DE UNA SUPERFICIE MATERIAL (EL MAPEO O TRAZADO DEL DISEÑO)

Después de la entrada de datos y de haber revisado los errores, procedemos a añadir **textura** a las superficies (es decir, **mapear** o **trazar**). En esta operación se utilizan los realces, en los cuales se añade color y textura a las superficies exteriores. Este proceso de colocar realces sobre las diferentes superficies exteriores se refiere a un **“mapeo de textura”**. Diversos materiales son utilizados en el exterior de la construcción tales como teja o azulejo, piedras y vidrio. Estas texturas son creadas a mano, leyendo en una memoria con un scanner, o tomando imágenes de una pantalla. Esta doble dimensión de datos entonces se traza sobre cualquier superficie como un realce.

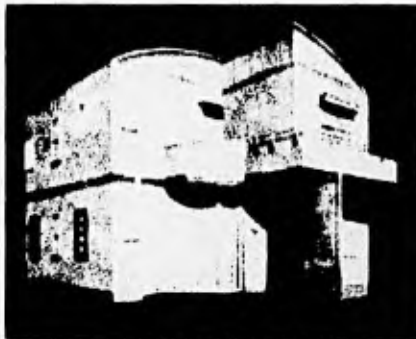
V.1.4 COMPLETANDO LA ESTRUCTURA

Una vez que hemos completado el trazado de texturas, el exterior de la construcción está terminado. Entonces en este momento es cuando debemos checar los errores en el trazado. También debemos buscar los colores que no combinen y otras incongruencias. Entonces es aquí cuando el trabajo de **crear una perspectiva** comienza. En el momento en que estamos checando las imperfecciones, también debemos checar estar pensando en como pueden ser usadas o como pueden ser presentadas estas perspectivas, es decir, que en este punto es en donde queremos obtener una comprensión total de la imagen de la perspectiva.

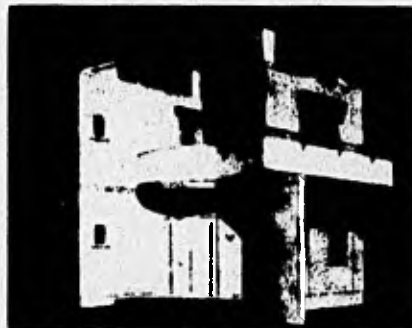
Las siguientes Figuras muestran cada una de las fases del método de producción.



El modelado (Esqueleto de alambre)



La interpretación



El trazado



Completando la estructura

V.1.5 ADICIONANDO LOS DETALLES

Después de terminar la estructura, se añaden los detalles. Aquí es donde se colocarán árboles u otros detalles, (tal como se muestra Figura V-1) tomando en cuenta que entre el tamaño de la estructura y el plantío de arbustos debe ser el correcto. Los objetos en el primer término deben ser interpretados claramente con aquellos lugares más lejanos, borrosos y tenues. Así, se conseguirá una imagen con una forestación apropiada y con una triple dimensionalidad.

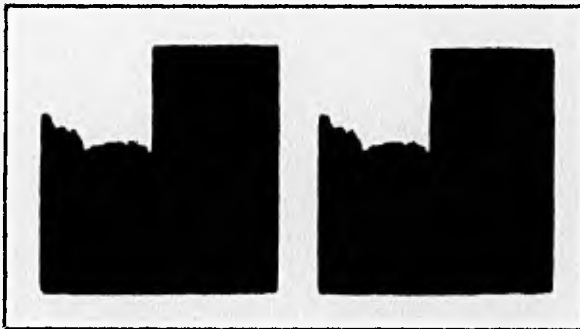


FIGURA V-1
Adición de los detalles

TECNICAS PARA DAR DETALLES

En el "Método de Producción" descrito anteriormente, vimos por etapas como se procede a partir de los datos dimensionales completos para terminar la estructura exterior de una construcción. Como se pudo ver, el proceso de mapeo o trazado es esencial para la creación de las perspectivas arquitectónicas en la **Graficación por Computadora**.

En vista de las perspectivas dibujadas, el estilo del artista se determina por medio del material utilizado tal como la brocha y el uso del color. Aquí es donde se presenta el aspecto atractivo del tipo de perspectiva que se dibuja. Por ejemplo dos personas que utilizan el mismo Hardware y Software para producir una perspectiva de los mismos materiales, cada una de ellas creará trabajos únicos, ya que los pensamientos y sentimientos de todas las personas se expresan en forma diferente. Aquí, el lápiz y la brocha son remplazadas por la computadora pero no reduciendo el poder y el sentimiento expresivo del artista. Lo que no debemos olvidar es que en las perspectivas dibujadas a mano hay estilos de brocha y color, mientras que en las perspectivas dibujadas por computadora existen técnicas de expresión. Tanto mayor dominio uno tiene de estas técnicas, tanto mayor capacidad se tiene de crear perspectivas por computadora originales.

Otra cosa que debemos recordar tanto en la pintura como en las perspectivas, es el balance, ya que este es de gran importancia.

Creación De Diseños Con Vidrio

Tratar de dibujar el vidrio moldeado en un monitor, se reduce el grado de la brillantez de la luz solar. En este ejemplo (Figura V-2), se podrá observar que el brillo decrece gradualmente. Posteriormente se trata de representar el grado de la rugosidad o de la aspereza de éste, y finalmente, el vidrio moldeado comenzará a tomar una apariencia realista añadiendo hasta cierto punto una transparencia. En la figura siguiente, el color del vidrio es azul verde pálido, de esta manera el espectador podrá percibir inmediatamente su calidad y entenderá lo que representa.

Creación De Diseños Con Metal Perforado

Recientemente el metal perforado ha ganado popularidad como un material arquitectónico. el cual ya es usado en diversas aplicaciones. Con una pequeña experimentación, es posible crear una superficie de metal perforado en una perspectiva realizada por computadora. En la Figura V-3 se muestra un trazado transparente, en el cual el color ((denso)) se muestra como transparente y el color delgado llega a ser opaco. Para simular el metal perforado se usa un color gris básico y los agujeros se crean utilizando un color obscuro (negro), es decir, cuando el trazado aparece como hoyos perforados. Esta es una manera de crear una superficie metálica perforada.



FIGURA V-2



FIGURA V-3

Utilización Del Vidrio Claro

En la vida cotidiana podemos distinguir fácilmente entre el vidrio transparente y el vidrio opaco. Si podemos ver lo que está al otro lado del vidrio, entonces, el vidrio es transparente. En la **Graficación por Computadora** podemos aumentar el nivel de transparencia del vidrio a través del comando "grado de transparencia" pero es importante no olvidar dibujar los objetos que se ven a través del vidrio. Esta operación es llamada "**Técnica de Trazado de Reflejo**", utilizada para añadir otras imágenes a la superficie del vidrio. Con eso se pueden añadir reflejos de cosas tales como árboles, edificios, el cielo y nubes. La Figura V-4 ilustra un ejemplo.

Utilización De Superficies De Bronce

Simplemente cambiando el color del vidrio transparente, se crea una superficie totalmente diferente. En este ejemplo (Figura V-5), el color del vidrio se cambia de azul verde a ámbar.



FIGURA V-4



FIGURA V-5

El Reflejo De La Luz Del Sol En El Vidrio

A diferencia de la luz del día (luz de color blanco), el ocaso contiene luz naranja, luz roja y mezcla de luz ultravioleta. En este ejemplo (Figura V-6) se suprimió la luz fuerte y se ajustó el color, esto no es una técnica particularmente difícil, pero nos ayuda a crear una impresión natural en la perspectiva. Por otra parte, los efectos naturales no han sido utilizados mucho en las perspectivas dibujadas a mano. Esta técnica, es por lo tanto, una consideración adicional a la perspectiva de la **Graficación por Computadora**.

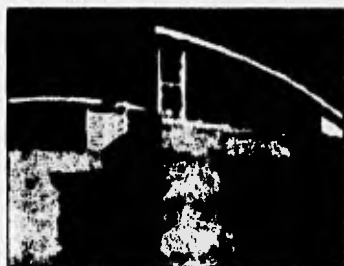


FIGURA V-6

Paredes Creadas Por Medio De Bloques De Vidrio

Una de las capacidades convenientes de la **Graficación por Computadora** es la técnica del diseño. En la **Figura V-7** se muestra el diseño de una pared creada por medio de bloques de vidrio colocados uno sobre otro hacia arriba. La función que permite una superficie larga se crea a partir de la repetición o colocación en capas de una forma repetidas veces, esta es la función **"patrón"** Aquí un patrón de bloques de vidrio ha sido creado de antemano con la intención de simular una superficie de bloques de vidrio, para después, cortarlo al tamaño apropiado y por último trazar la textura. Es así como la pared de bloques de vidrio se termina entonces

El comando **"patrón"** es utilizado también para crear superficies de teja o azulejo y piedra. En breve, ésta será una técnica utilizada a menudo en la **Graficación por Computadora**.

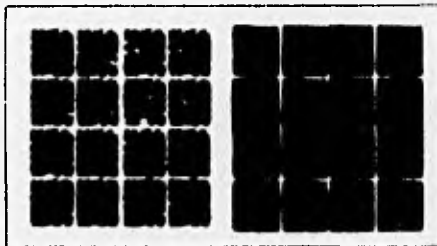


FIGURA V-7
Diseño creado mediante bloques de vidrio

Creación De Ventanas Tipo Persianas

Las **ventanas tipo persianas** como se ilustran en la **Figura V-8** se encuentran frecuentemente en las casas. Tales marcos de las persianas no parecen particularmente complejos a simple vista, pero en realidad el largo de la hoja de vidrio una encima de la otra forman un patrón complejo. Para la creación del diseño y los propósitos prácticos, es importante decidir si la atención va a dibujarse o a mostrarse para las piezas de los objetos o de los objetos como un todo. Los datos dimensionales para cada una de las persiana serán una tarea que consuma mucho tiempo. Pero

dibujarse o a mostrarse para las piezas de los objetos o de los objetos como un todo. Los datos dimensionales para cada una de las persiana serán una tarea que consuma mucho tiempo. Pero para ahorrar tiempo se puede indicar simplemente que el marco se crea mediante muchas capas particulares de piezas de vidrio

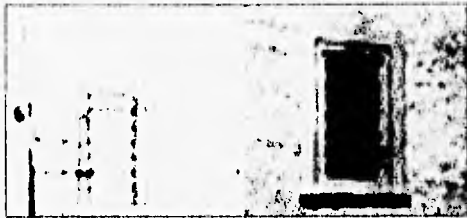


FIGURA V-8
Ventanas tipo persianas

Creación Del Registro De Ventilación

El registro de la ventilación es sólo un pequeño punto en la construcción, pero en las perspectivas de gráficas por computadora su manera de representar llega a ser un asunto de interés. Si observamos la forma del registro cuidadosamente veremos que éste no solamente es esférico, sino que parte de la esfera está en la sombra. Con el fin de crear esta característica comenzaremos asumiendo que el registro es una forma esférica con una esfera más pequeña embebida en ella. Entonces para la creación del diseño nos concentraremos en la sección que está primero en la sombra. En la Figura V-9, por ejemplo se oscureció la porción mellada, acarreado con esto que las tres dimensionalidades de la figura se reconozcan fácilmente.



FIGURA V-9
Registro de ventilación

Paredes Con Textura De Piedra

Para dibujar imágenes de un catálogo, uno puede obtener **superficies con textura de piedra real**. Pero no hacer simplemente uso de los datos dibujados para crear superficies, sino también pensar en la perspectiva, en los procesos y en el complemento de datos para que se ajuste dicha perspectiva. Para que se emplee este diseño de superficies con textura de piedra el diseño básico de la superficie se debe trazar en su totalidad sobre una rejilla. Existen varias texturas de superficies empedradas. Para este ejemplo, la Figura V-10 solo muestra la textura de tipo granito. Primeramente se debe crear la rejilla y el diseño para la teja empedrada, después se decora el muro completo.

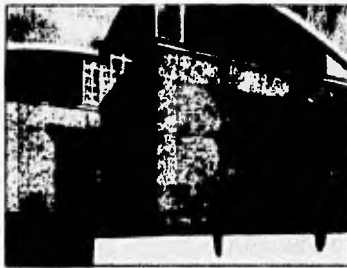


FIGURA V-10
Textura de tipo granito

Paredes Con Textura De Teja o Azulejo

Las superficies decoradas con teja o azulejo proveen un ejemplo típico de decorado. Uno crea decorados básicos y después los utiliza para diseñar con ellos. Pero esto no es todo, hay que poner los azulejos, la teja, etc. Otra técnica, la cual puede ser usada es la técnica de **“Enderezar el Trazo”**. En este caso, más bien que la descripción del azulejo, es la descripción de la rejilla entre los azulejos. Hay además rejillas profundas y poco profundas. La profundidad de las líneas entre los azulejos tienen un efecto real sobre la apariencia de la estructura. Con líneas profundas parecen sombras, el contraste es alto y la estructura toma una apariencia muy clara. Con líneas

poco profundas la superficie de la construcción mostrará una superficie plana. Esto es algo muy difícil de expresar en las superficies de perspectivas dibujadas a mano. Pero con el método de "Enderezar el Trazo" utilizado en las perspectivas de la *Graficación por Computadora* se hace esto posible. Los datos en los cuales se basa este método se expresan como sombras en las áreas cóncavas, con las cuales uno puede lograr cierta profundidad o un efecto poco profundo, así como la expresión deseada para las superficies de la construcción. La imagen siguiente (Figura V-11) es un ejemplo del método "Enderezar el Trazo".



FIGURA V-11
Aplicación del método "Enderezar el trazo"

La Figura V-12 muestra el diseño una vez ya terminado de la casa que se tomó como ejemplo.



FIGURA V-12
Diseño terminado de la casa habitación

V.2 METODO DE PRODUCCION PARA EL DISEÑO INTERNO

Uno de los principales propósitos de las perspectivas interiores es el de mostrar la distribución del interior como la mejor disposición del mismo interior y de las habitaciones para el constructor. Esta es una de las barreras difíciles que deben superarse en las perspectivas interiores.

En otras palabras, la **Perspectiva Interior** debe comprender la sensibilidad del lugar en los mismos conceptos altamente subjetivos y ambiguos y debe además ser efectiva como una perspectiva. Esto es porque los creadores de perspectivas interiores no sólo deben tener técnicas sino también buena sensibilidad.

Con respecto a esto las perspectivas realizadas por medio de las computadoras son una herramienta posible de expresión. Entre las principales ventajas de las computadoras gráficas están sus habilidades para calcular la iluminación, para presentar la textura de las superficies y de ofrecer un rango extremadamente amplio de colores.

Los pasos a seguir en la **producción de perspectivas interiores** son:

- **Datos Dimensionales De Entrada (El Modelado)**
- **Verificación De Los Datos De Entrada (La Representación)**
- **Adicionando Texturas (El Mapeo o Trazado del Diseño)**
- **Completando El Interior**
- **Adicionando Los Detalles, Terminado De La Perspectiva.**

Este es el orden a seguir en el proceso de exteriores e interiores. Sin embargo, en los interiores el detallado es demasiado importante. Generalmente hablando, un espacio está compuesto de pisos, paredes, techo y columnas pero para dar un terminado pleno, se

necesitan detalles como sofás, mesas, plantas y cortinas, las cuales deben ser colocadas alrededor para dar sentido a la habitación y traer vida a la perspectiva.

V.2.1 DATOS DIMENSIONALES DE ENTRADA (EL MODELADO)

Los **datos dimensionales** se toman desde los planos del piso hasta los planos del área, ya que por ejemplo, para saber la posición precisa de una ventana desde el plano del piso y para saber el ancho de esta, se debe consultar el plano del área.

En una situación como esta donde hay indirectamente luz en el techo, aún se puede estar requiriendo para los datos de entrada segmentos de los planos detallados del área.

Además con las perspectivas interiores la posición de la ventana no es sólo el factor, otro dato particular es la iluminación, la cual debe ser checada necesariamente ya que es un factor importante en el ambiente de cualquier interior. No sólo su colocación y su figura, sino también el **tipo de iluminación** (directa o indirecta) debe ser checada.

V.2.2 VERIFICACION DE LOS DATOS DE ENTRADA (LA REPRESENTACION)

De la misma manera como en las perspectivas exteriores los datos de entrada se usan para desarrollar un diseño de la perspectiva interior, pero hay que tener en cuenta que en este tipo de perspectivas es mucho más importante cuidar los detalles, ya que el interior se monta sobre el piso, muros y techo.

Después de meter y checar cuidadosamente los datos del piso, muros y techos así como la mejor posición de las ventanas y la luz, se debe crear una vez más las formas de la imagen del interior

total. Estas actividades pueden referirse como para la verificación de los datos dimensionales originales.

V.2.3 ADICIONANDO TEXTURAS (EL MAPEO O TRAZADO DEL DISEÑO)

El terminado de cada superficie es llamado **Trazado**, recientemente muchos interiores de casas tienen entarimado o alfombrado, y paredes barnizadas. Las texturas del piso pueden mostrarse fácilmente con colorantes. Para revestir las paredes, la proporción especular debe suprimirse completamente para lograr un terminado totalmente plano.

Con respecto al diseño de las paredes, para lograr un terminado rugoso o áspero, se recomienda el enyesado. Al crear el diseño, se debe incluir sombreado sobre la superficie terminada para dar una expresión clara del diseño. El terminado de la pared se llevará a cabo por medio de la iluminación.

V.2.4 COMPLETANDO EL INTERIOR

Después de representar todas las superficies, y después de haber checado si hay o no errores en los bordes y coloridos, el interior se considera como terminado.

V.2.5 ADICIONANDO LOS DETALLES, TERMINADO DE LA PERSPECTIVA

Cada espacio interior tiene una parte que cumple una función específica. En este ejemplo mostramos una habitación donde la sala funciona como una sala y el comedor como un comedor. Cada espacio interior es un escenario donde cada habitante juega diferentes papeles.

A continuación se muestran las figuras que representan los pasos descritos anteriormente.



El modelado



Verificación de los datos de entrada



El mapeado o trazado



Terminado de la perspectiva

TECNICAS PARA DAR DETALLES

Como expresamos antes no sólo el piso, paredes techo y columnas del interior sirven para la expresión de los detalles, el sofá, mesa, sillas, etc. determinan también el éxito o el fracaso de las perspectivas.

La distancia en una perspectiva interior es más cerrada que en una perspectiva exterior, lo cual significa que cada elemento debe tener una textura real. Si hay una copa media llena de agua, esta debe parecer una copa, Si el sofá es suave, debe verse suave. Las cortinas, sillas, plantas y los cuadros, todos contribuyen al ambiente de la habitación.

En la **Graficación por Computadora** es conveniente tener todos los elementos necesarios ya introducidos dentro de una base de datos para que puedan ser utilizados una y otra vez en algún trabajo, y sus aplicaciones tengan un aspecto práctico en las perspectivas de la **Graficación por Computadora**.

Un punto que debe hacerse notar acerca de las perspectivas interiores hechas por computadora, es considerar el diseño de la iluminación.

En un interior el alumbrado se llama **"La iluminación de la habitación"**. Una manera de iluminar una habitación es hacer que la luz surja del techo y de las paredes, en la **Graficación por Computadora** es posible controlar la brillantez (la claridad y obscuridad) y el color (fluorescente o incandescente) de la iluminación. Por lo tanto, sin alterar los datos dimensionales uno puede mostrar la habitación en la noche o en cualquier hora del día. Esto es extremadamente útil para ver exactamente que tipo de atmósfera producirá el alumbrado en la habitación.

Sin duda en el futuro muchas de las aplicaciones de las perspectivas de la **Graficación por Computadora** serán más desarrolladas por aquellos involucrados en trabajos de gráficas por computadora.

Dispositivos De Iluminación

Los dispositivos de luz baja son frecuentemente agradables en la iluminación cuando se representa esta clase de iluminación. Dos factores deben de ser considerados, uno es la forma y el color del dispositivo y el otro es el efecto de la iluminación, en otras palabras, es la expansión de la luz desde el dispositivo y su interacción con el espacio entero.

Algunas luces bajas expanden toda la iluminación a lo largo de la habitación, otras concretan luz en una área localizada, mediante el estudio del diagrama del dispositivo de luz baja, uno puede ver como son logrados estos efectos. En breve, esto es hecho por la lámpara o foco colocado en el techo, ya que este tipo de luces son empotradas en él (Figura V-13).



FIGURA V-13
Dispositivo de luz baja

La claridad indirecta describe una fuente de luz oculta la cual lanza iluminación en el piso, muros y techo. Esta clase de iluminación es suave y es esencial para observar exactamente como los dispositivos de luz están colocados en una posición oculta y a la vez analizar la iluminación de la habitación. Las clases de lámparas que pueden ser utilizadas son las fluorescentes o incandescentes, ya que estas lanzan una luz diferente, lo que significa que el ambiente de la habitación va a ser distinto. Esto es además agradable para jugar con los colores en la iluminación indirecta. En la Figura V-14 una luz azul se ha utilizado



FIGURA V-14

Los datos dimensionales para los muebles no son de gran problema. Pero las dimensiones de alguna clase de muebles, como los de arte Rococo (es decir, arte Luis XV) toman un gran tiempo para entrar y pueden ser llamados **diseños difíciles para perspectivas de Graficación por Computadora**. Por lo tanto, para las perspectivas realizadas por computadoras los muebles modernos son preferidos.

En el siguiente ejemplo (Figura V-15) se muestra un sofá simple, pero esta clase de sofá puede estar hecho de piel o de tela. Con una cubierta de piel uno podría obtener una superficie reflectora. Sin embargo con tela, el brillo sería controlado por un efecto suave.

Existen dos clases de alfombrado, muro a muro y por medio de tapetes. Cuando son utilizados tapetes se incluyen alfombras de la más alta calidad como las de China, ya que ellas tienen

muchos diseños y son los productos artísticos de larga historia. Con la **Graficación por Computadora** estos objetos pueden ser fácilmente reproducidos (Figura V-16).

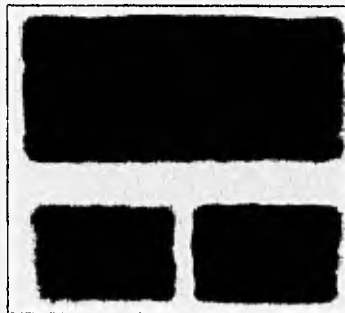


FIGURA V-15
Sofá



FIGURA V-16
Diferentes tipos de tapetes

Existen varios materiales de entarimado, la siguiente Figura muestra un interior terminado con largas tablillas.



Materiales de entarimado

La Figura siguiente muestra los interiores ya terminados.



CONCLUSIONES

Como pudimos observar, la **Graficación por Computadora** tiene una gran utilidad en la Arquitectura, ya que gracias a ella se reducen los tiempos en el diseño de edificios u otro tipo de inmuebles, en este caso, en una casa habitación tomada como ejemplo.

Mediante el uso de la **Graficación por Computadora**, se diseñó la parte externa e interna de la casa siguiendo el mismo método para las dos partes, el cual consiste en los pasos siguientes :

- El Modelado.
- Verificación de la Entrada de los Datos.
- El mapeado o Trazado de la Perspectiva.
- El Terminado de la Perspectiva o del Diseño.

Con uso de la **Graficación por Computadora**, nosotros podemos jugar adaptando diferentes diseños a cada parte de la casa o bien, cambiando los colores cuantas veces queramos, el terminado del diseño depende del diseñador mismo, ya que aquí se plasma la combinación de los colores y muebles que el diseñador le adapte a cada habitación.

CONCLUSIONES

En el transcurso del presente trabajo, pudimos darnos cuenta de lo que es la **Graficación por Computadora** y la importancia que ha adquirido en los últimos 10 años.

Las gráficas por computadora, como observamos, datan desde los primeros tiempos de las computadoras, es decir, cuando los sistemas eran capaces de producir dibujos primitivos mediante la impresión de líneas, encontrándose la ventaja de convertir grandes volúmenes de datos numéricos en imágenes significativas y fácilmente interpretables, todo esto fue lo que le dio gran importancia a la **Graficación por Computadora**. (Capítulo II).

Para la creación del software gráfico lo más importante es la utilización de algoritmos eficientes que permitan una respuesta rápida del sistema al dispositivo controlado por el usuario, ya que como se recordará, una imagen se describe por medio de una **secuencia de órdenes de graficación** elementales tales como **Mover y Dibujar**. Estas órdenes son interpretadas por el algoritmo para conversión por barrido correspondiente, que calcula a su vez los **pixeles correctos para iluminar**. A este conjunto de órdenes gráficas que representan la imagen se conoce como **Archivo de Presentación**. (Capítulo III).

Con esto, podemos concluir que la velocidad con la que se producen **figuras gráficas**, depende de la capacidad de la computadora y la eficiencia de los algoritmos utilizados, todo esto **basado** en un hardware especial. Cualquier imagen es creada mediante la colocación de **subimágenes** que la componen en posiciones específicas dentro del sistema de **Coordenadas Mundiales** (que son las coordenadas cartesianas que un usuario encuentre conveniente), a este proceso lo conocemos

como **Montaje**. Una subimagen se crea por separado requiriendo el uso de transformaciones geométricas para posicionar la subimagen correctamente en la imagen. Estas transformaciones son la base para la creación de cualquier imagen (Capítulo IV).

Actualmente podemos observar que la **Graficación por Computadora** ha adquirido una gran importancia en todas las áreas tales como : la ingeniería, la medicina, la administración, la economía y muchas otras, que si hiciéramos una lista de ellas nos llevaría mucho espacio nombrarlas; todas ellas hacen últimamente de la **Graficación por Computadora** una herramienta indispensable para su desarrollo.

Nosotros para la realización de este trabajo sólo escogimos una de esas áreas que es la **Arquitectura**, ya que desde nuestro punto de vista, por el diseño de edificios, casas, hospitales, oficinas y sobre todo en la decoración interior y exterior de estos inmuebles (Capítulo V), podemos apreciar el diseño ya terminado o hacer cualquier combinación que nosotros nos imaginemos hasta quedar convencidos y nos guste el diseño.

Por último, podemos decir que la **Graficación por Computadora** continúa en desarrollo, facilitando con esto la reducción de problemas en todas las áreas y con la diversidad de innovaciones de hardware y software, encontrarán muchas aplicaciones al igual que reducciones en costo y tamaño.

GLOSARIO

- A -

ACERCAMIENTO (Zooming) : La herramienta de zooming permite cambiar rápidamente el tamaño de todo o parte de la imagen en la que se está trabajando.

ALGORITMO : Es una secuencia de pasos diseñados para resolver un problema o ejecutar un proceso tal como dibujar una curva desde una serie de puntos de control.

APLICACION : Es un programa de software especialmente diseñado para las necesidades del usuario o la especificación del uso de un programa. Las aplicaciones gráficas son usualmente diseñadas para habilitar que el usuario manipule datos o imágenes; o para crear imágenes.

ATRIBUTO : Son las propiedades de las primitivas de salida tal como el color de la intensidad, estilos de líneas, estilos de texto y modelos de llenado de áreas.

- B -

BIT : Palabra corta de dígito binario ("binary digit"). Indica la unidad más pequeña de información que es almacenada en una memoria digital. Los dígitos binarios indican dos valores, y en memoria se representan como '0' inhabilitado y '1' habilitado.

BORDES : En graficación son uno o más vectores definiendo una porción de un objeto. En el procesamiento de imagen son un grupo de valores determinados para ser la división de la línea entre una y otras imagen.

BUFFER CON PROFUNDIDAD : Es un método de imagen-espacio que se utiliza comúnmente para eliminar superficies ocultas. Básicamente este algoritmo verifica la visibilidad de superficie un punto a la vez.

BUFFER DE ESTRUCTURA O DE RENOVACION : Es el almacenamiento de la renovación en los sistemas de rastreo con rastreador. También es llamado mapa de bits.

BYTE : Es una agrupación de dígitos binarios adyacentes. (bits) operados en la computadora por una unidad. El tamaño más común de un byte contiene 8 bits.

- C -

CAD : (Diseño Asistido por Computadora). Término genérico que comprende cualquier actividad en que una computadora interviene en el diseño.

CIRCULACION (Scrolling) : La circulación permite moverse a partes del dibujo que no son visibles en la pantalla, sea por el tamaño del dibujo o por el nivel de acercamiento (Zoom).

CLIPPING (Corte) : Es una operación bidimensional o tridimensional que reduce el número de dibujos calculados hecho en el CPU por la eliminación de un objeto o porciones de los objetos.

COMPILADOR : Es un lenguaje del alto nivel que permite traducir un programa escrito, a un programa que la máquina puede entender (lenguaje máquina).

COMPUTADORA : Es una máquina electrónica que junto con otros dispositivos como lo son el teclado, monitor y otros es capaz de realizar por una secuencia de pasos, una serie de objetivos que son fijados por el usuario.

CONVERSION DE RASTREO : Es el proceso de la generación de información de pixeles en el buffer de estructura del programa de aplicación.

COORDENADAS ABSOLUTAS : Son las referencias coordenadas en el comando polyline, es decir, que estos valores son las posiciones reales de puntos en el sistema coordinado en uso.

COORDENADAS DE DISPOSITIVO O DE PANTALLA : Son las coordenadas que utilizan los dispositivos de salida particular.

COORDENADAS DE DISPOSITIVO NORMALIZADOS : Son las coordenadas del sistema entre las coordenadas mundiales del usuario y las coordenadas de dispositivos físicos del sistema en la vista de tubería.

COORDENADAS DE SISTEMAS : Es el sistema matemático particular de ejes en los cuales los puntos y la líneas pueden ser rastreados por virtud de sus distancias desde el origen, o posicionados en varios pasos en la vista de tubería. También son la ubicaciones 'x' y 'y' (para 2-D) y 'x', 'y' y 'z' (para 3-D) y son llamados puntos coordinados.

COORDENADAS MUNDIALES : Los programas de aplicaciones definen imágenes en un sistema de coordenadas mundiales. Este puede ser cualquier sistema de coordenadas cartesianas que un usuario encuentre conveniente. Las imágenes definidas en coordenadas mundiales son mapeadas por el sistema de gráficas en coordenadas de dispositivo.

CPU (Unidad Central de Proceso) : Es el cerebro de la computadora y sirve de almacenamiento a los datos e instrucciones que van a hacer procesadas además de ejercer cierto control sobre estos. Desarrolla operaciones aritméticas, lógicas y de control. Controla la entrada y salida de información en los dispositivos periféricos.

CRT : Tubo de Rayos Catódico.

- E -

ESCALACION : Es una transformación para alterar el tamaño de un objeto.

- G -

GRAFICACION BIDIMENSIONAL : Son todas aquellas imágenes que se presentan con altura y anchura y son presentadas en el plano cartesiano conformado por los ejes 'x' y 'y'.

GRAFICACION POR COMPUTADORA : Puede definirse como la creación de imágenes gráficas por medio de una computadora.

GRAFICACION TRIDIMENSIONAL : Es el despliegue de objetos y escenas con altura, anchura y profundidad. La información es calculada en las coordenadas de sistemas que representan los ejes 'x', 'y' y 'z'.

- H -

HARDWARE : Es la parte física que forma a las computadoras, es decir, los circuitos eléctricos y los dispositivos periféricos.

- I -

ICONO : Son símbolos gráficos que se parecen a la opción de procesamiento que deben representar y son usados como opciones de entrada a muchos programas de computadora.

INTERFACE DE APLICACION : Es la interfaz para una librería de subrutinas para un lenguaje especificado que implementa niveles altos de funciones gráficas.

- L -

LIBRERIAS GRAFICAS : Es un grupo de herramientas para programas de aplicación e interfaces en un programa de interface de aplicación. Las librerías gráficas usualmente incluyen una definición de una serie de primitivas y funciones.

LINEA DE RASTREO : Es cada una de las líneas horizontales de pixeles que dibujan una figura en la pantalla, ya que ellas se dibujan desde el buffer de estructura, una hilera a la vez.

- M -

MARQUESINAS (Marques) : Una marquesina es una ventana de uso especial que se utiliza para seleccionar elementos con el mouse.

MEMORIA RAM (Random Acces Memory) : Este tipo de memoria a diferencia de la memoria ROM, pierde su información con la falta del suministro eléctrico. Además permiten escribir, leer o modificar los datos ya grabados.

MEMORIA ROM (Read Only Memory) : Es una memoria de sólo lectura y son utilizadas para grabar los programas de uso general

MODELADO : Es la creación y manipulación de la representación de un sistema.

MODELADO DE SISTEMA : Es cualquier reproducción individual de un sistema, estos pueden definirse gráficamente o puramente descriptivos como un conjunto de ecuaciones que define las relaciones existentes entre los parámetros de un sistema

- O -

OBJETO : Es una simple imagen o modelo definido en un espacio bidimensional o tridimensional.

- P -

PASEO (Panning) : El paseo (panning) es una forma de moverse por un dibujo sin tener que circular. Cuando se pasea, la ventana del dibujo automáticamente circula al momento que se arrastran objetos por los extremos. Esto evita tener que marcar el objeto, utilizando barras de circulación, y después reelegir y continuar arrastrando.

PIXEL : Es cada una de las posiciones del buffer de estructura, también es llamado elemento de la figura.

PLANO : Es un espacio infinito definido por tres puntos que no se encuentren en la misma línea.

PLANO DE VISION : Es la vista determinada de una escena especificada por el usuario. Es la superficie sobre la cual se proyecta la vista de un objeto. También puede ser considerado como la película de una cámara que se ha posicionado y orientado para una toma determinada.

POLYLINE : Es el comando primitivo (o básico) de salida en un paquete de gráficas, para el trazo básico de puntos individuales y líneas.

POLYMARKER : Es el comando de caracteres más adecuado para la colocación de un carácter designado en una o más posiciones seleccionadas.

PRIMITIVA GRAFICA : Es la estructura básica de una imagen.

PROCESAMIENTO DE IMAGENES : Es la técnica de graficación que se usa para producir despliegues visuales a partir de fotografías o exploraciones de televisión.

PROGRAMA : Es una serie de instrucciones que realizan una función determinada.

PROYECCION EN PARALELO : Es el proceso de una imagen proyectada desde el volumen visto en tres dimensiones hacia una gráfica bidimensional con proyección en paralelo. También son los objetos en alguna distancia desde el punto de referencia visual manteniendo su tamaño aparente bajo la proyección en paralelo. Estas proyecciones son ortogonales en el plano bidimensional.

PUERTA DE VISION : Es el área rectangular en el dispositivo de despliegue en el cual se coloca la ventana.

PUNTO DE REFERENCIA VISUAL : Es la posición elegida por el usuario en coordenadas mundiales para establecer las coordenadas de visión, esta posición sirve como el punto de referencia visual. También se dice que es la ubicación de la vista en un espacio tridimensional.

RASTER : Es una rejilla rectangular que contiene elementos de imagen o píxeles. Los datos gráficos para ser desplegados son almacenados por el buffer de estructura.

REFLEXION : Es la transformación que produce una imagen de espejo de un objeto.

REJILLAS (Grids) : Las rejillas son una capa base de puntos que no se imprimen y que se entrecruzan dividiendo la pantalla en sectores. Se pueden mover objetos por la rejilla de manera que sea posible posicionar objetos exactamente. El movimiento por la rejilla es como tener una rejilla magnética que puede retener el objeto en esta y mantenerlo ahí.

RESOLUCION : Es el número máximo de puntos que pueden exhibirse sin superposición en un CRT y esta depende del tipo de sustancia fosforescente que se utilice y de los sistemas de enfoque y de flexión.

ROTACION : Es la transformación de los puntos de un objeto situado en trayectorias circulares.

- S -

SEUDONIMO : Este provoca la distorsión en la representación de objetos trazados en un rastreador.

SISTEMAS DE COORDENADAS DE VISION : Es el sistema de coordenadas que se define para establecer el plano de visión.

SISTEMA OPERATIVO : Son los programas que le indican a la computadora como debe de trabajar, es decir, es el traductor entre la máquina y el usuario.

SOFTWARE : Es la parte de la computadora que esta compuesta por programas en general, el sistema operativo, los compiladores, etc. Es la parte que hace funcionar a la computadora.

- T -

TRASLACION : Es el movimiento en línea recta de un objeto, de una posición a otra.

TRANSFORMACION : Es un cambio hecho en el tamaño, localización u orientación de un objeto.

- V -

VENTANA : Es el área rectangular que se especifica en coordenadas mundiales.

BIBLIOGRAFIA

- **COMPUTER GRAPHICS
ARCHITECTURAL RENDERING
A SUPER - REALISTIC COLLECTION**
DE. MACGRAW HILL.
PAG 18 - 54

- JONH L. BRADBERRY
COMPUTER GRAPHICS ENVIRONMENTS
ED. SAMS PUBLISHING
IRA. EDICION USA

- **AN INTRODUCTION TO COMPUTER GRAPHICS CONCEPTS FROM
PIXELS TO PICTURE**
ED. SUN MICROSYSTEMS
IRA. IMPRESION 1991
PAG 47 - 54, 99 - 102

- BERGER MARC
GRAFICACION POR COMPUTADOR CON PASCAL
DE. ADDISON-WESLEY IBEROAMERICANA
USA 1991
PAG. 396

- MORRIS L. ROBERT Y ECKHOUSE RICHARD H
SISTEMAS DE MINICOMPUTADORA
ED. PRENTICE HALL INTERNATIONAL
FEBRERO, 1982

- ARECHIGA R. CORCHADO. J.
FUNDAMENTOS DE COMPUTACION
ED. LIMUSA

- MORA JOSE LUIS, MOLINO ENZO
INTRODUCCION A LA INFORMATICA
ED. TRILLAS
4A. EDICIÓN
ENERO, 1985

- F. ROGERS DAVID, ADAMS J. ALAN
MATHEMATICAL ELEMENTS FOR COMPUTER GRAPHICS
ED. MC GRAW HILL
USA 1976
PAG. 239

- HEARN DONALD, BAKER M. PAULINE
COMPUTER GRAPHICS
ED. PRENTICE-HALL, INC., ENGLEWOOD
CLIFFS, NEW JERSEY
PAG. 352

- FOLEY J.D., VAN DAM A.
FUNDAMENTALS OF INTERACTIVE COMPUTER GRAPHICS
ADDISON-WESLEY PUBLISHING COMPANY
USA 1984
PAG. 664