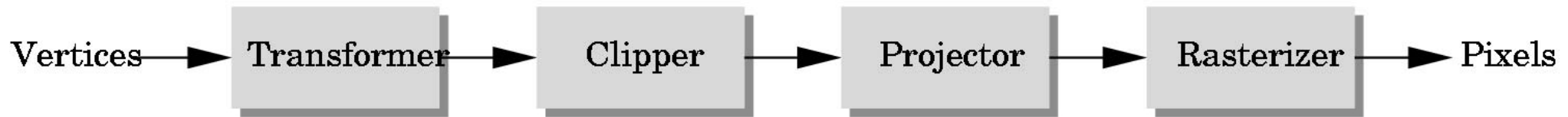


Formación de Imágenes



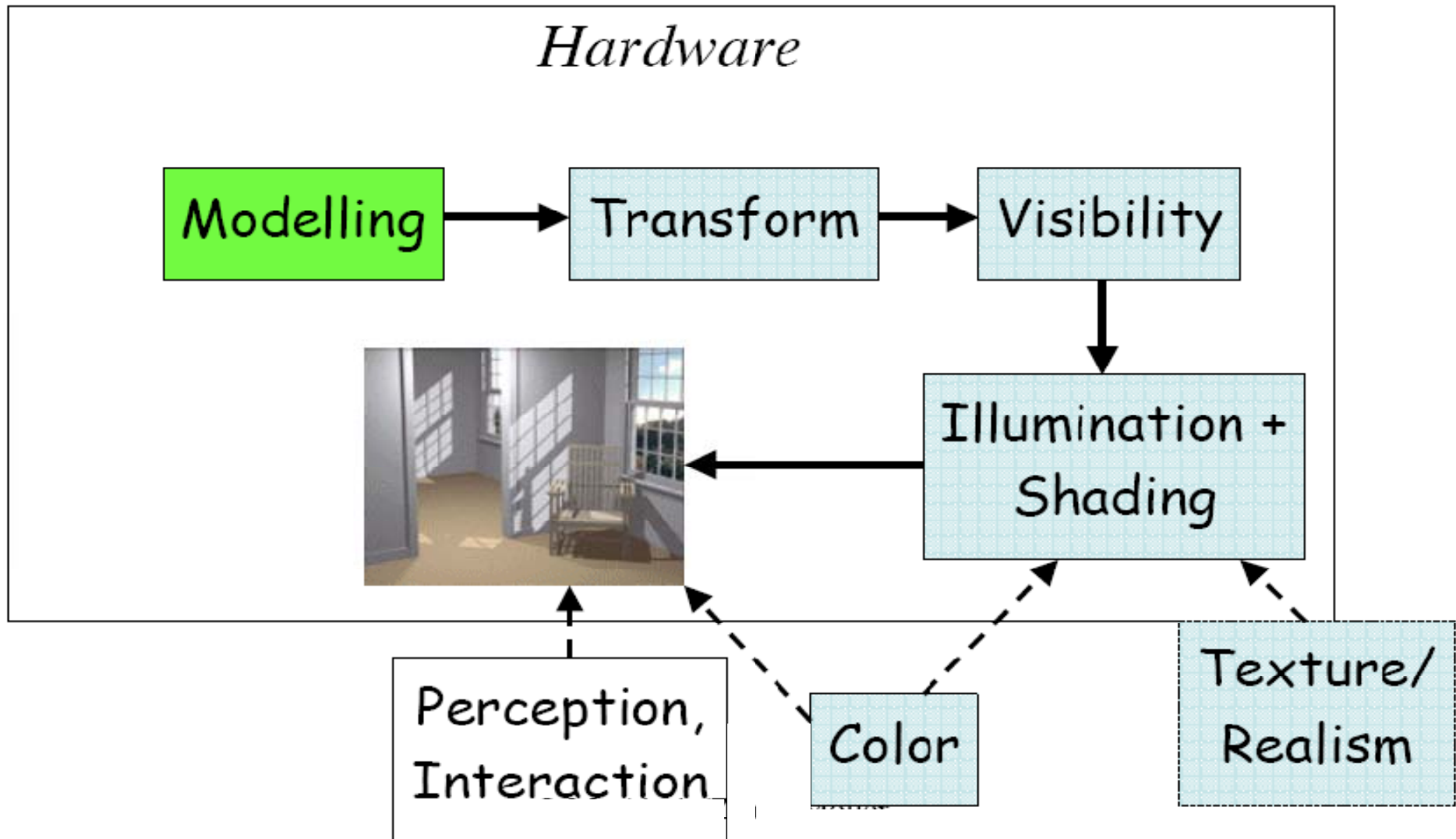
- Pipeline del proceso de formación de imágenes



Renderizado



□ Pipeline del proceso de renderizado



Pipeline de Gráficas



- ❑ Del modelo a la imagen
- ❑ Pipeline define los diferentes procesos por donde debe pasar la data en el sistema
 - ❑ Los procesos pueden ser ejecutados en paralelo

Procesador de vertices



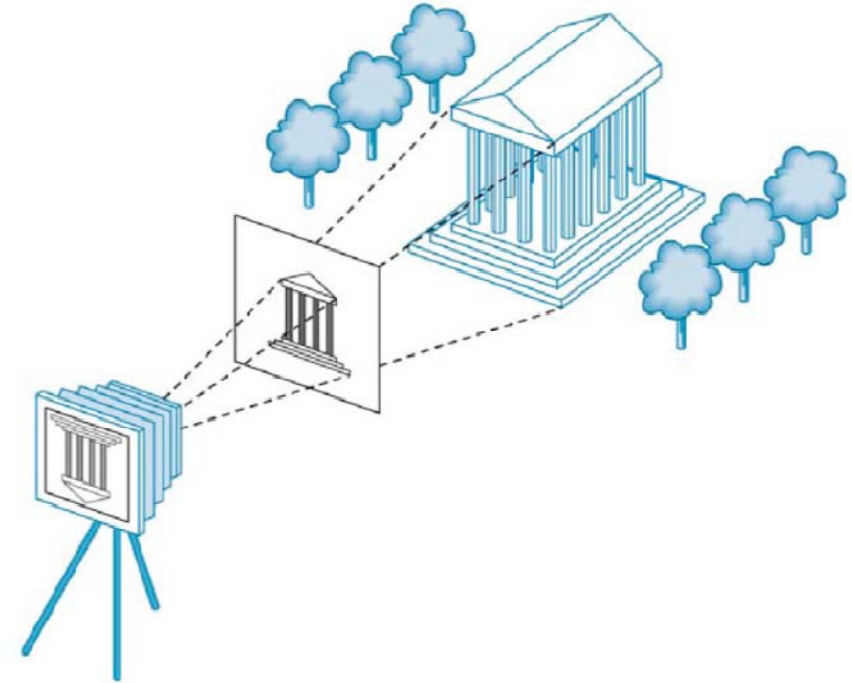
- ❑ Operaciones de vertices definen las primitivas
- ❑ Funcionalidades principales
 - ❑ Transformar las coordenadas
 - ❑ Calcular el color en cada vertice (sombreado)
- Geometria de los objetos son definidos en su propio sistema de coordenadas luego transformados al sistema de coordenadas del mundo (transformación de modelado – MODELVIEW)
- ❑ Luego los objetos son transformados del sistema de coordenadas del mundo al sistema de coordenadas de la cámara (transformación de visualización y de proyección)



Recorte y ensamble de primitivas



- ❑ Modelar un campo finito de visión
- ❑ Eliminar los objetos que están fuera del área del volumen de recorte
- ❑ Las transformaciones se deben hacer por cada primitiva
- ❑ La salida es un conjunto de primitivas cuya proyección aparece en la imagen



Rasterizado



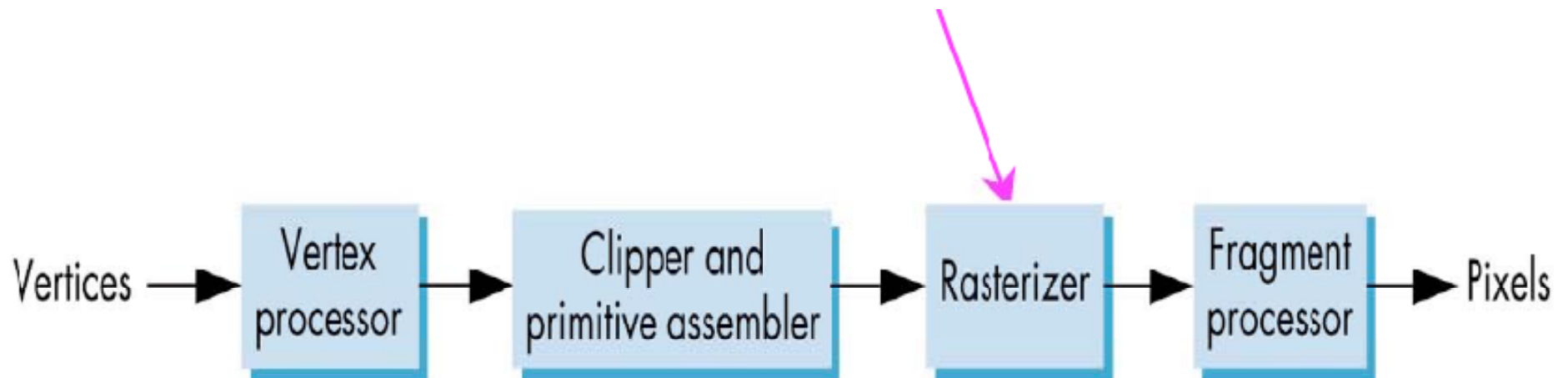
- ❑ El proceso de rasterización convierte una primitiva en un conjunto de fragmentos
- ❑ Un fragmento almacena por pixel la información asociada a la primitiva, luego es usada para determinar si el fragmento contribuye al color del pixel
 - ❑ Ubicación del pixel (en el framebuffer)
 - ❑ Profundidad
 - ❑ Color, textura
 - ❑ Valor Alpha
 - ❑ Ventana ID



Proceso de Rasterización de primitivas



Dibujo o rasterización de las primitivas
Comenzamos por líneas y polígonos



Líneas en OpenGL



```
glColor3f(1.0, 1.0, 1.0);  
glLineWidth(2.0);  
glBegin(GL_LINES);  
    glVertex2f(-0.5,  
              -0.5);  
    glVertex2f(-0.5, 0.5);  
    glVertex2f(0.5, 0.5);  
    glVertex2f(0.5, -0.5);  
glEnd();
```

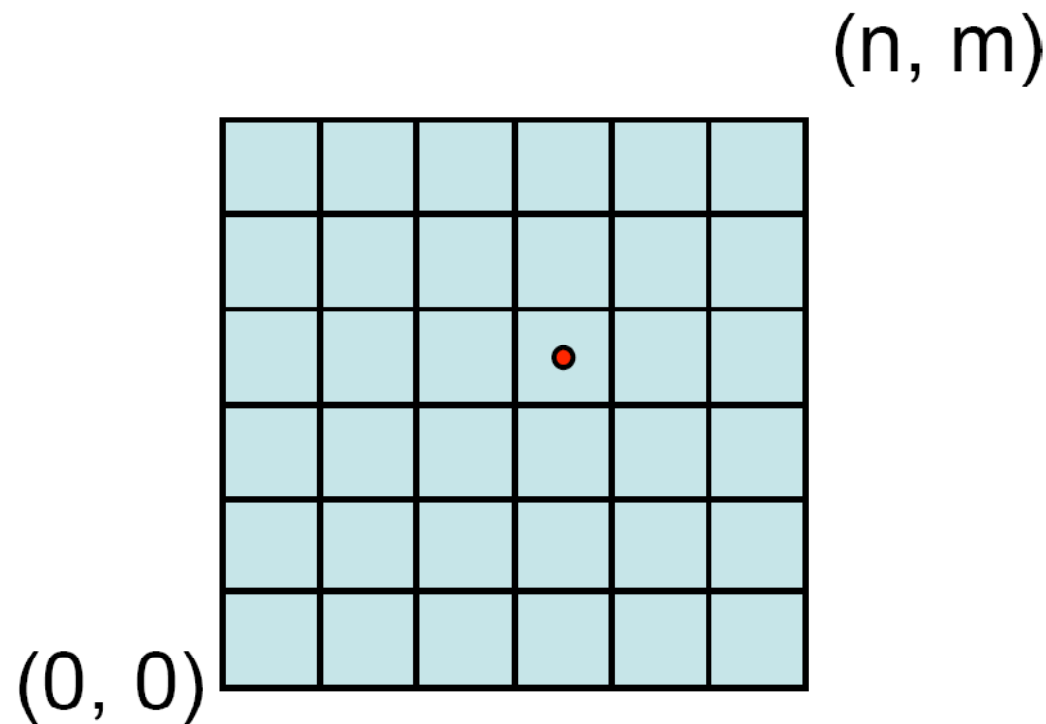
glBegin(GL_LINE_STRIP);

glBegin(GL_LINE_LOOP);

No existe GL_CIRCLE



- ❑ Transformaciones, recorte y proyección ya se hicieron
- ❑ Primitivas a rasterizar están en la pantalla
- ❑ Pantalla maneja coordenadas 2D de forma





- ❑ Poligonos
- ❑ Curvas explicitas
 - ❑ Líneas, etc.
- ❑ Curvas y superficies paramétricas
- ❑ Otras
 - ❑ Sistema de particulas, puntos, fractales

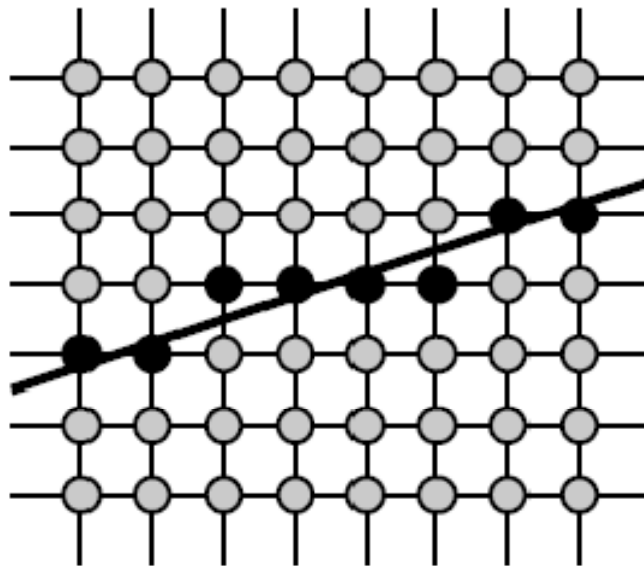
Conversión de una línea



- Dada la función explícita de una recta

$$y = mx + n$$

- Que pixel dibujar?



Conversión de una línea

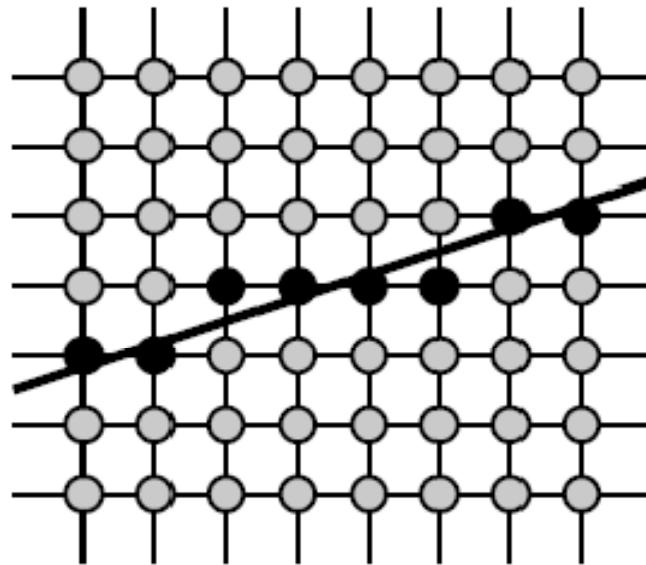


- ❑ Propiedades deseadas
 - ❑ Brillo constante (indiferentemente de la longitud u orientación)
 - ❑ Un pixel por columna ($-1 \leq \text{pendiente} \leq 1$), un pixel por fila (pendiente $>+/-1$)
 - ❑ Tan cerca como sea posible de la línea recta
 - ❑ Dibujado tan suave como sea posible (anti-aliasing)
 - ❑ Dibujar tan rápido como sea posible

Conversión de una línea



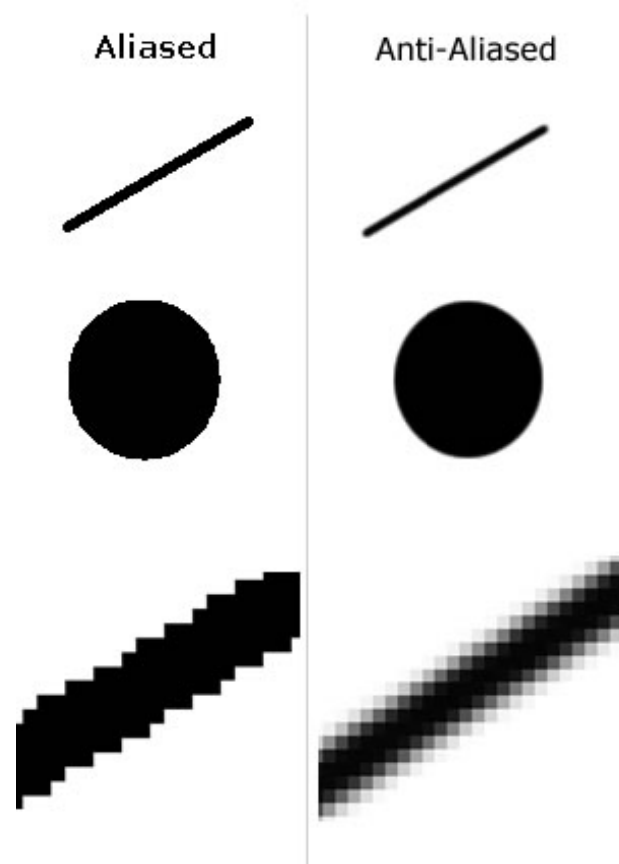
- ❑ Propiedades deseadas
 - ❑ Un pixel por columna ($-1 \leq \text{pendiente} \leq 1$), un pixel por fila (pendiente $> + / - 1$)
 - ❑ Tan cerca como sea posible de la línea recta



Conversión de una línea



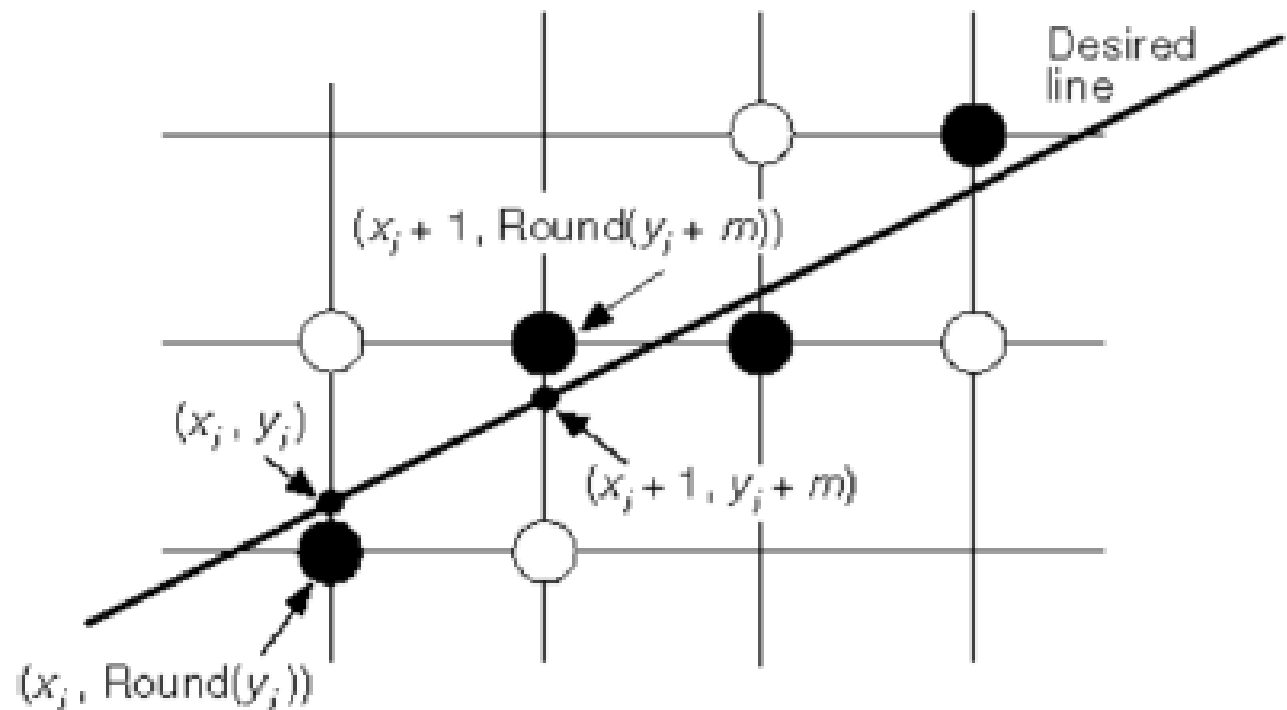
- ❑ Propiedades deseadas
 - ❑ Dibujado tan suave como sea posible (anti-aliasing)



Incremental (básico)



- ❑ Encontrar la pendiente $m = \text{rise/run}$ or $y = mx + b$
- ❑ Incrementa en x en 1 (x_i), calcula $y_i = mx_i + b$
- ❑ Asignar 1 al pixel $(x_i, \text{round}(y_i))$
- ❑ Simple pero ineficiente
- ❑ Desventajas
 - ❑ punto flotante
 - ❑ Multiplicación
 - ❑ redondear



Incremental DDA



- ❑ Elimina la multiplicación

$$\begin{aligned}y_{i+1} &= mx_{i+1} + b \\ &= m(x_i + 1) + b \\ &= y_i + m\end{aligned}$$

- ❑ digital differential analyzer (DDA) – resuelta por métodos numéricos
- ❑ Desventajas
 - ❑ Punto flotante– floating point values (m,y)
 - ❑ redondear
 - ❑ Casos especiales se pueden hacer más rápido

Bresenham



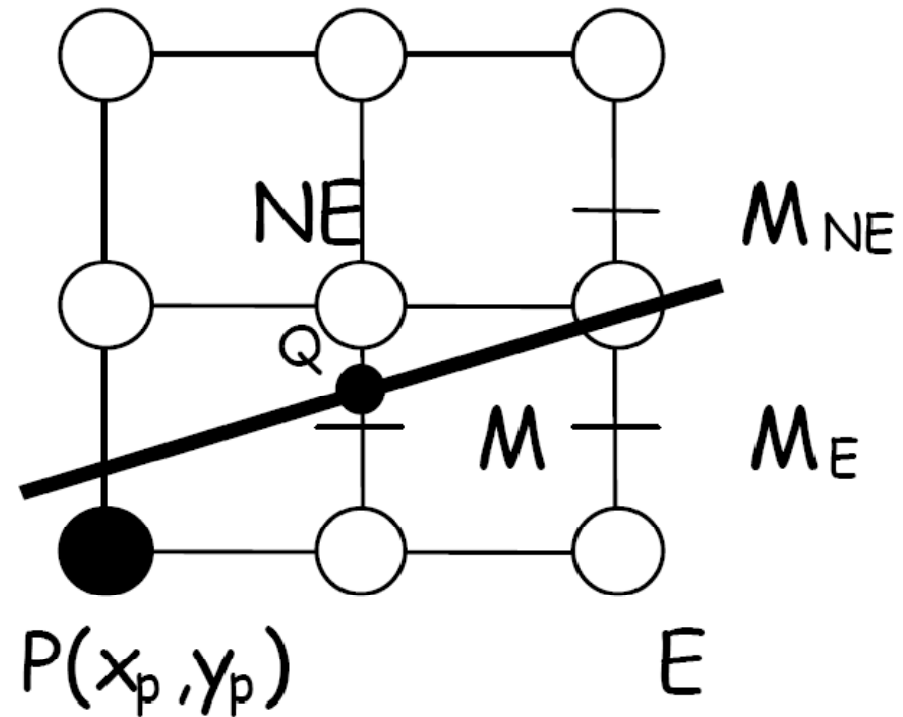
- ❑ Caso especial del algoritmo de punto medio
 - ❑ Solo operaciones de aritmética de enteros
 - ❑ No se hace redondeo
- ❑ Proveer la aproximación que mejor se ajuste a línea recta minimizando el error (distancia) a la línea recta
- ❑ Pendiente $0 \leq m \leq 1$

Bresenham



□ Idea:

- Si $Q > M$ selecciona NE
- Si $Q < M$ selecciona E
- Si $Q = M$ selecciona cualquier otro
- Error $< 1/2$



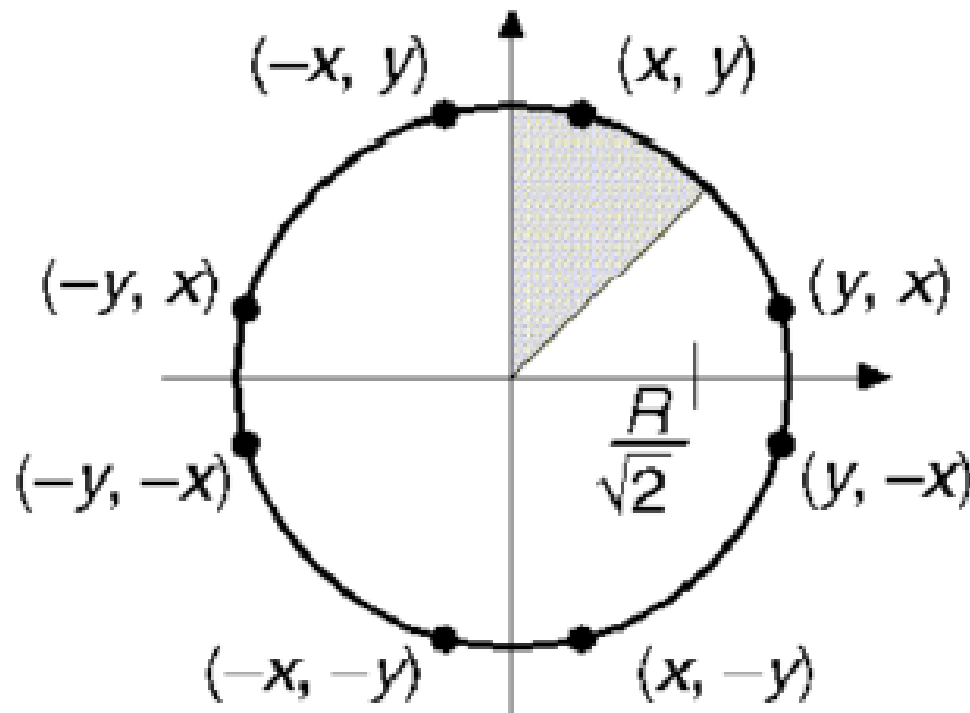
Trazado de Circulos



- Círculos o elipsoides

$$y = \sqrt{R^2 - x^2}$$

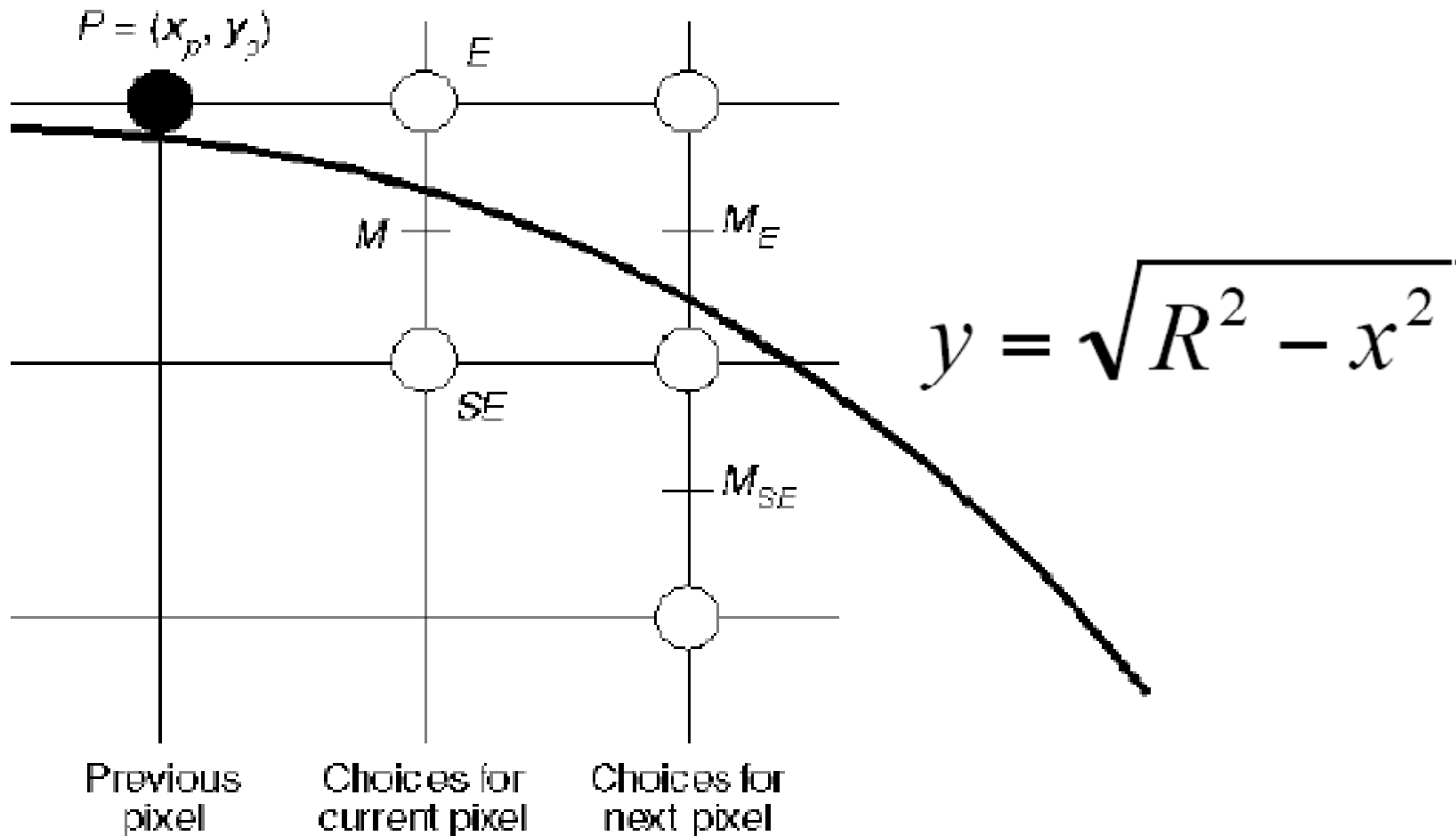
- Dividir el círculo en varios cuadrantes



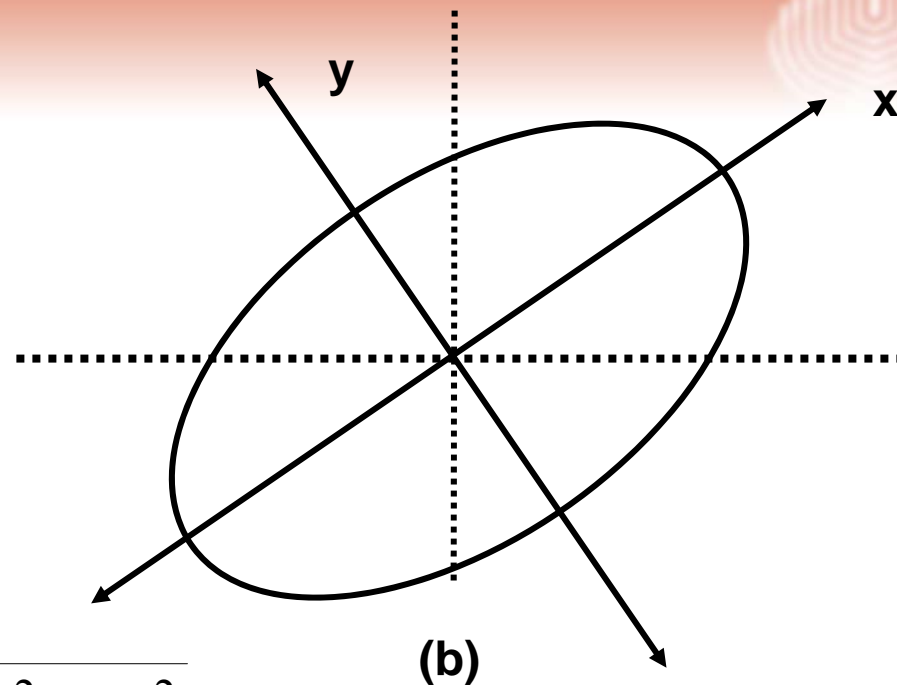
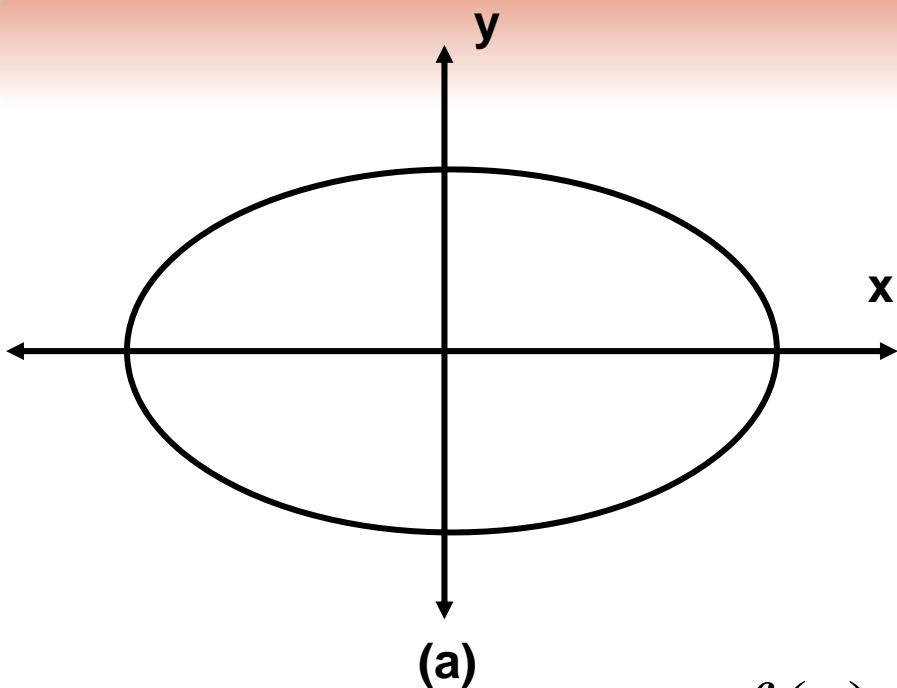
Trazado de Circulos



- Algoritmo de Medio punto
 - Trazar un arco en un cuadrante y dibujar los otros por simetria
 - Seleccionar entre E y SE en cada avance



Trazado de Elipses



$$y = f(x) = \frac{b}{a} \sqrt{a^2 - x^2}$$

