

C. PROCESADOR DIGITAL DE SEÑALES PDSP14188

A continuación se presenta un resumen de las características del PDSP14188 que han sido empleadas en la configuración del bloque encargado de realizar las operaciones de convolución.

C.1 Operación Básica

Este DSP realiza una suma ponderada de cada pixel dentro de una ventana $N \times N$. Cada valor de pixel de la ventana es multiplicado por un valor numérico denominado peso, o simplemente coeficiente, y su suma es puesta a la salida. En las aplicaciones prácticas, se emplean valores positivos de coeficientes para obtener promedios de imagen o positivos y negativos para detectar los bordes. La ventana se desplaza sobre la imagen de vídeo digitalizada hasta cubrirla por

completo permitiendo operaciones en tiempo real.

C.2 Precisión en la Salida

Con pixels de ocho bits, y una ventana de 8x8, es posible aumentar el valor de la suma acumulada hasta un tamaño de palabra máximo de 22 bits empleando un solo dispositivo. Con 16 bits/pixel, y un tamaño de ventana de 8x4 -el máximo posible-, la suma puede aumentar hasta 29 bits. El PDSP16488 actualmente permite el crecimiento de la palabra hasta un máximo de 32 bits conectando dispositivos en cascada sin peligro de producir un overflow.

En una aplicación particular la salida deseada podría estar desplazada dentro esos 32 bits, la posición dependerá de los valores de los coeficientes usados, esto causa problemas físicos para saber cuales de los pines de salida se conectarán al sistema. Para superar este problema, el DSP contiene un multiplicador de salida, o Control de Ganancia, el cual permite alinear el resultado final -al bit más significativo hasta un máximo de 32 bits- de salida. La inclusión de un multiplicador, más que un simple desplazador, permite definir una ganancia con mucha precisión.

Los 16 bits más significativos del resultado ajustado quedan disponibles sobre pines de salida, incluyen un bit de signo.

C.3 Saturación de Salida

Si la salida del DSP está manejando un monitor de video, los pixels negativos darán resultados erróneos. Una opción es proporcionada para forzar todos los resultados negativos al valor cero, los que serán interpretados como puntos negros por el monitor. Al mismo tiempo los resultados positivos, los

que producen overflow en el Control de Ganancia, son forzados a un valor de saturación igual al número más positivo, esto es el blanco. En este modo el bit de signo a la salida será siempre cero, y no debería conectarse al convertidor D/A que conduce al monitor.

Otra opción diferente fuerza ambos valores de overflow, los positivos y los negativos, a los valores de saturación correspondientes a sus respectivos valores máximos, pero en la escala negativa los resultados negativos permanecen válidos. Existe también un flag -bandera de aviso- que avisa cuando el Control de Ganancia ha producido un overflow, el cual puede ser conectado al host PC o a otra unidad anfitriona para ajustar el Control de Ganancia al adecuado de manera que este problema se solucione rápidamente.

C.4 Salida Binaria

El DSP contiene un comparador de 16 bits que permite comparar la salida del Control de Ganancia con un valor previamente programado. Un flag permite al usuario determinar si el resultado está por encima o por debajo del valor programado que se encuentra almacenado en un registro interno.

C.5 Multiplicador

El DSP contiene 16 multiplicadores, dispuestos en matrices de 8x8, cada uno produciendo un resultado de 16 bits. Internamente el reloj del pixel suministrado por usuario puede ser multiplicado por dos o por cuatro, lo que junto a la arquitectura propia, permite operar más rápido y utilizar cada multiplicador varias veces dentro de un mismo periodo de reloj de pixel. Esto aumenta el número efectivo de multiplicadores que están disponibles para el

usuario (Figura C.1) , de 16 a 32 ó 64 respectivamente. Esta arquitectura produce un empleo muy eficiente del área de la pastilla, y permite acomodar las líneas de retardo dentro del mismo dispositivo.

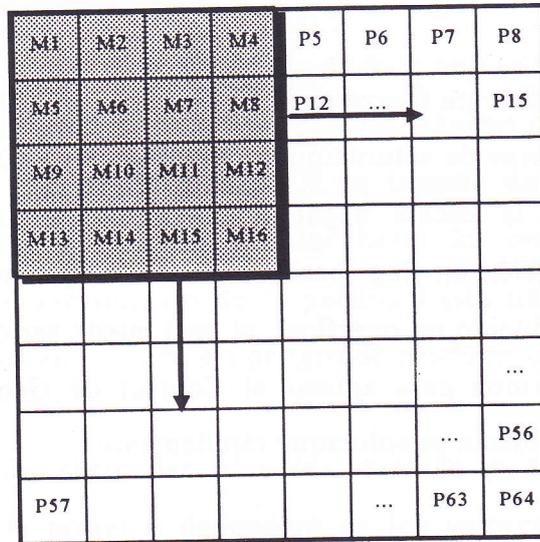


Figura C.1 Representación de los Multiplicadores

Los 16 multiplicadores son dispuestos en un array de 4 de profundidad por 4 de ancho, resultando un array efectivo de 4x8 o 8x8 con la opción multiciclo. El array multiplicador puede ser configurado para manejar palabras de 16 bits con signo; entonces el número efectivo de multiplicadores es dividido a la mitad.

C.6 Operación con líneas de Retardo

Una RAM interna separada en dos grupos se encuentra en el interior del dispositivo, y éstos pueden ser configurados para suministrar líneas de retardo para nivelar el tamaño adecuado del Convolver. Cuando un array de profundidad igual a cuatro con pixels de 8 bits, están disponibles cuatro líneas de retardo, y cada una puede ser programada para contener hasta un máximo de 1024 pixels. En un array de ocho de profundidad, o si es necesario emplear 16bits/pixel, cada línea puede contener hasta 512 pixels.

Si se desea implantar una máscara de 8x8, se deberán emplear dos convolvers de 8x4 para poder trabajar a 20 MHz, y tamaño máximo de línea. Esto se verá en la siguiente sección. La Figura C.2 ilustra las opciones disponibles. La primera línea retardo de línea en uno de los grupos puede ser opcionalmente cambiada dentro o fuera con una señal conectada a un pin de entrada. Esta característica se emplea para retardar el pixel de entrada cuando los datos son obtenidos de otro Convolver en un sistema de dispositivos múltiple, o también para soportar vídeo entrelazado. Las señales L7:0 pueden ser empleados como pixels de entrada o salida. Son configuradas como entradas a power-on para evitar posibles conflictos de bus, pero activando el bit de control de modo que pueden cambiarse a salidas. Estas líneas pueden ser empleadas para manejar otros dispositivos cuando se requiere de múltiples "convolvers".

C.7 Vídeo Entrelazado

Cuando se emplea vídeo entrelazado en tiempo real, una imagen o frame está compuesta de dos campos, con las líneas impares en un campo y las pares en el otro. En este caso se necesita añadir un *bloque adicional de retardo*, externo al sistema DSP, que recoja la información de las líneas adyacentes; por tanto el DSP necesita dos líneas de buses.

El bus que alimenta los pixels retrasados tienen una línea de retardo extra. Esto es usado sólo en el campo que contiene la línea más alta en cualquier par de líneas, y debe ser puenteado en el otro campo. Esto asegura que los campos previos siempre corresponden a la línea de arriba con relación a la actual, y evita la necesidad de cambiar la posición de los coeficientes de un campo al siguiente. La Figura C.3 muestra el traslado de líneas físicas a las posiciones de línea internas para un sistema constituido por un solo DSP.

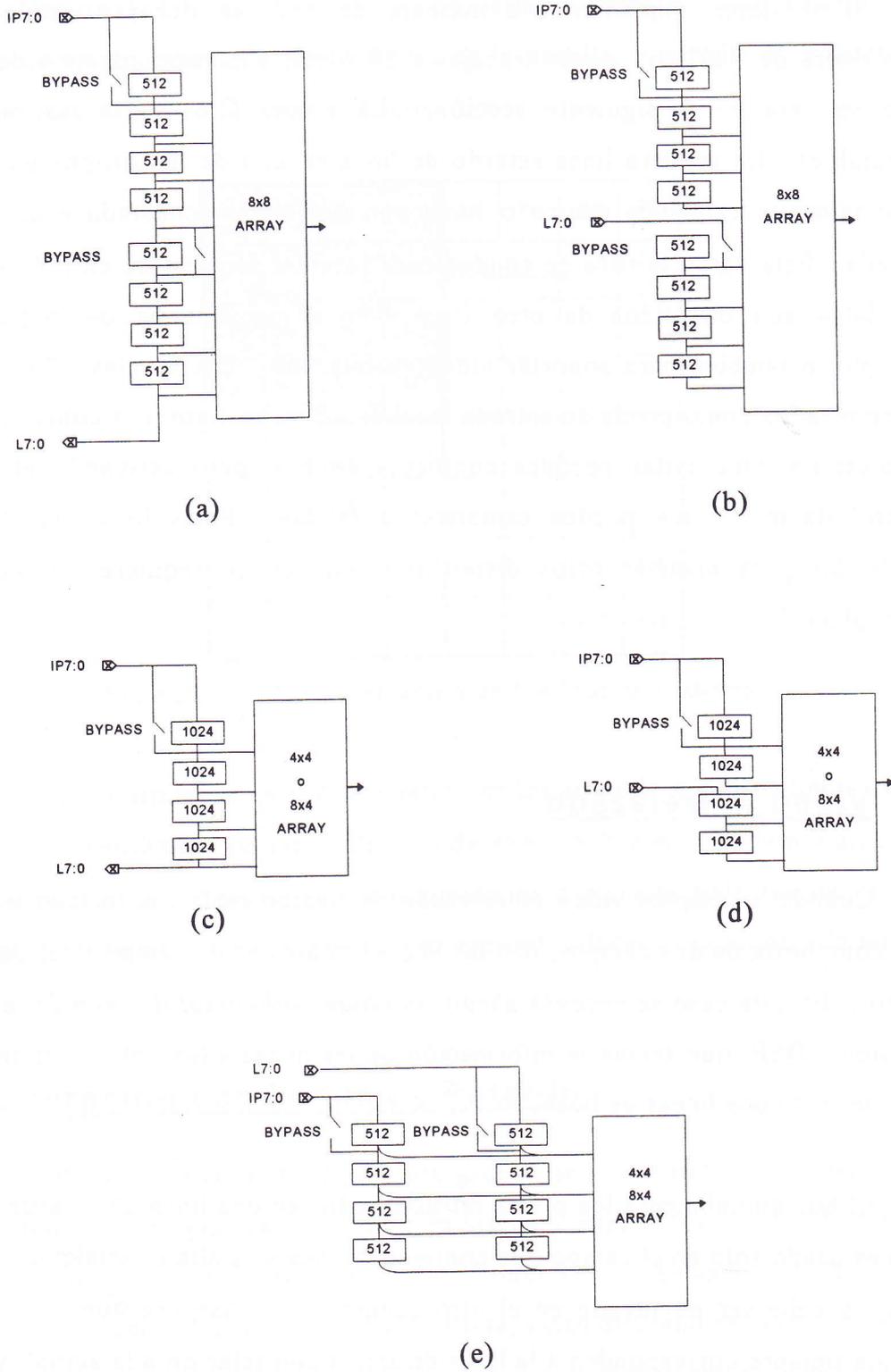


Figura C.2 Configuraciones para Retardos de Línea.

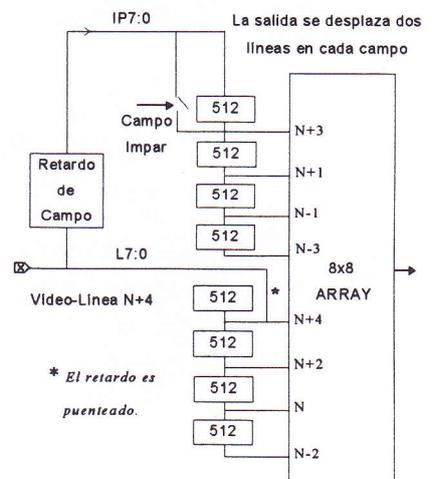
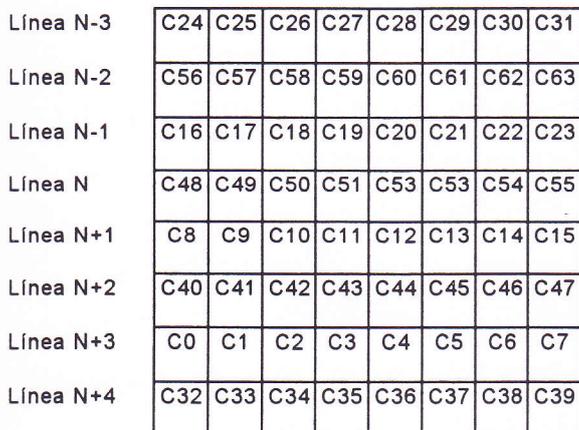
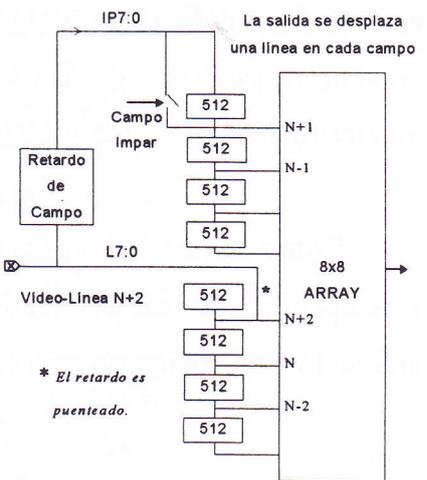
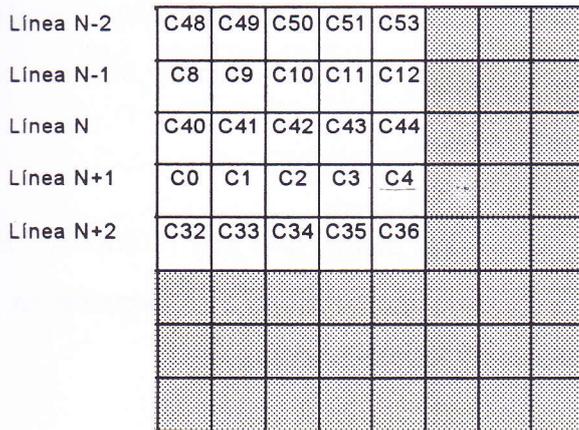
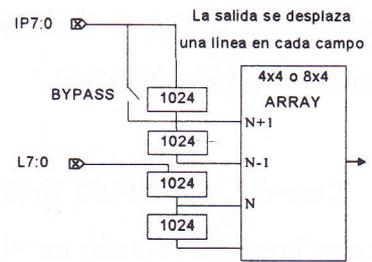
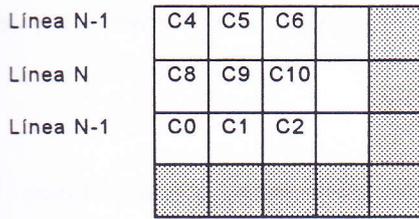


Figura C.3 Retardos de Dispositivo - Sistema Entrelazado.

La Línea N es la línea que está siendo convolucionada, la cual corresponde a una o dos líneas previas a la que se produce efectivamente según sean las dimensiones de la máscara.

Cuando la ventana requiere la implementación de cuatro o más líneas, la primera línea de retardo en el grupo alimentado por los pines L7:0 debe siempre ser puentado. Esta opción de puente está controlada por el Registro B, bit 7, internos, y no se ve afectada por la entrada conectada al pin de puenteo. Los coeficientes deben ser cargados en las localizaciones mostradas, las cuales nivelan la posición de las líneas trasladadas con los coeficientes no usados, mostrados sombreados, y cargados con ceros.

Como se puede ver, las posibilidades de este dispositivo son múltiples. En este apéndice se han destacado las características que influyen directamente en la aplicación planteada en esta tesis.

D. PRUEBAS Y EJEMPLOS

Para realizar las pruebas con imágenes reales se ha trabajado con la tarjeta digitalizadora de vídeo PSG8104-2, la cual está basada en un transputer T800. Asimismo, para la captura de imágenes se emplearon dos tipos de cámaras: la JVC TK-870E en color, y la Sony XC77CE en blanco y negro. Aunque se comenzó a trabajar con imágenes en color, las pruebas finales se hicieron con imágenes en blanco y negro, por ser éstas las que daban mejores resultados. En cuanto al software empleado, consiste en una librería básica proporcionada por el mismo fabricante de la tarjeta. Este software IPLIB trabaja en C 3L paralelo. También se empleó lenguaje C de Microsoft, Occam, y el entorno matemático MatLab. A continuación se presentan, a manera de ejemplos, algunas de las pruebas realizadas con imágenes de tonalidades de grises múltiples. Sus dimensiones son 160x160 pixels, lo que complica el trabajo del filtro.

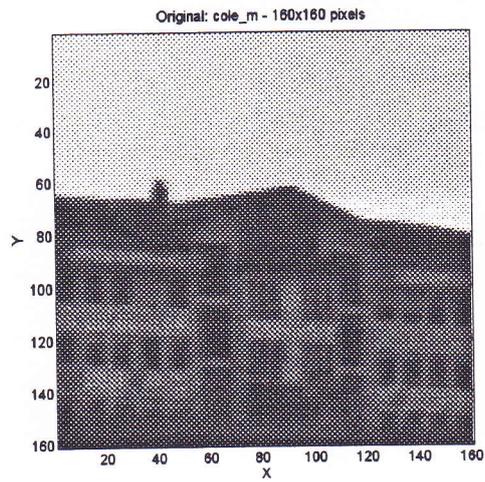
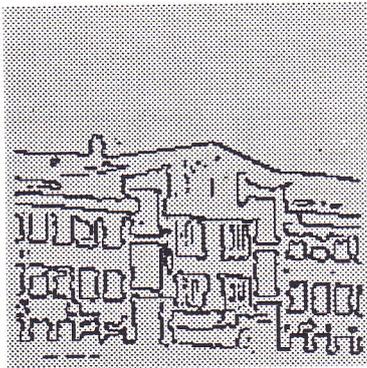
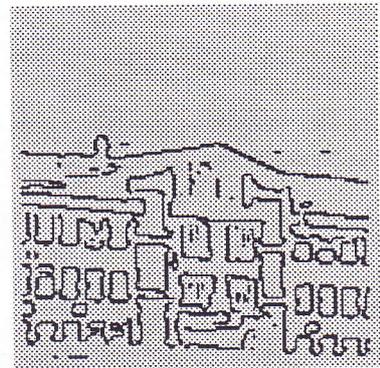


Figura D.1

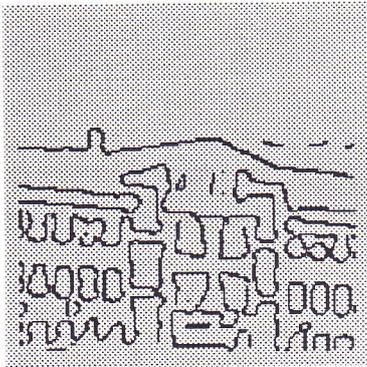
M6%2



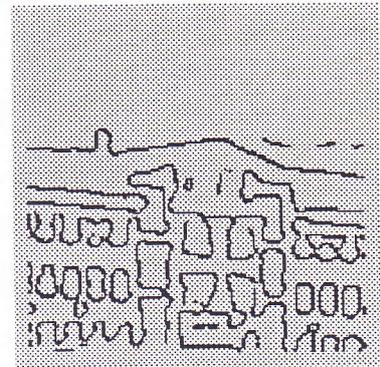
D1U601.8B2565k5K9



D2U163.2B2340k5K15



D2.5U459B1972k7K19



D3U433.5B1971k7K23

Figura D.2

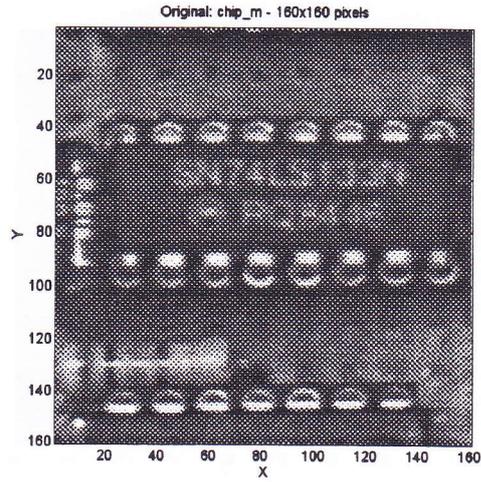
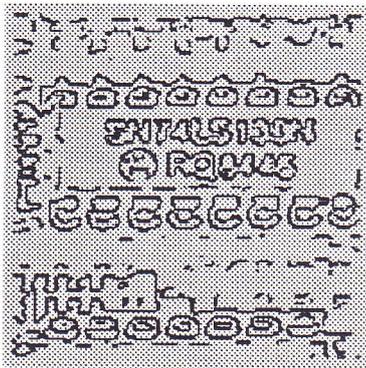
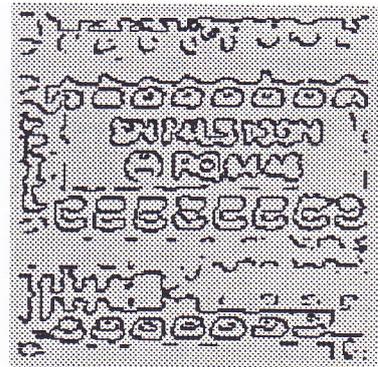


Figura D.3

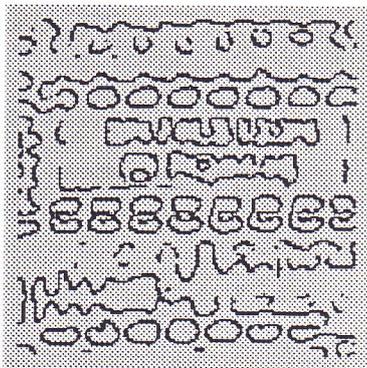
M6%5



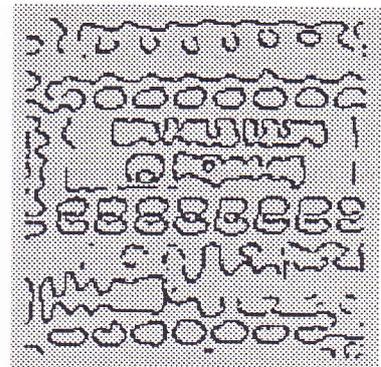
D1U1505B3613k5K9



D2U408B3532k5K15



D2.5U1148B3031k7K19



D3U1084B3029k7K23

Figura D.4

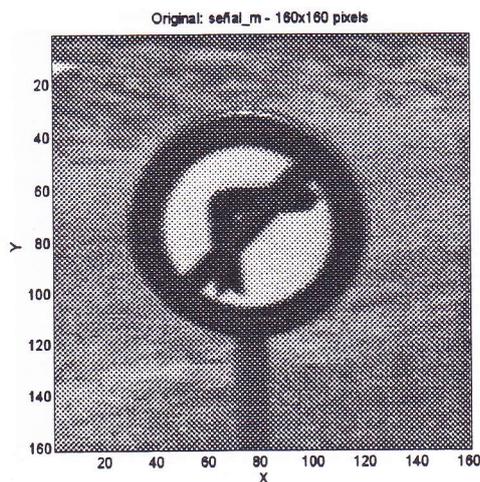
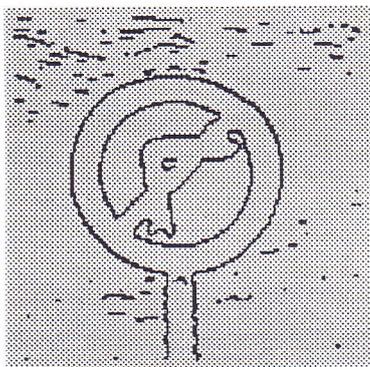
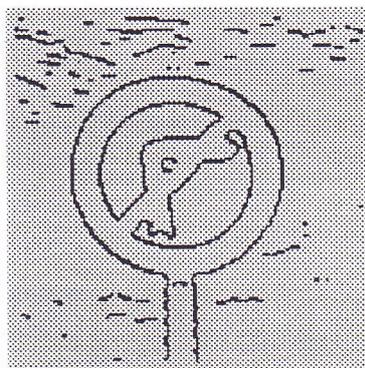


Figura D.5

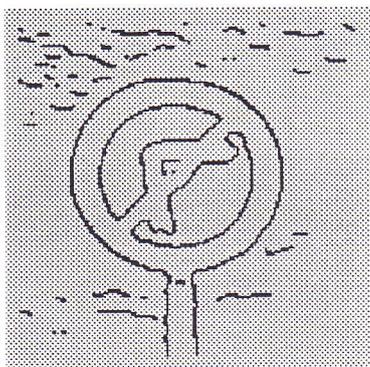
M6%5



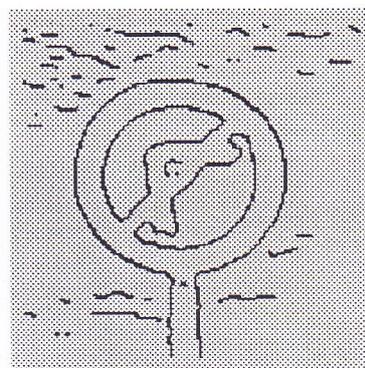
D1U1505B1190k5K9



D2U408B1220k5K15



D2.5U1148B1136k7K19



D3U1084B1137k7K23

Figura D.6

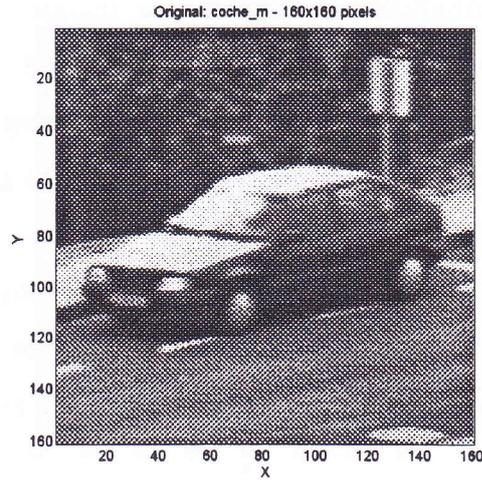
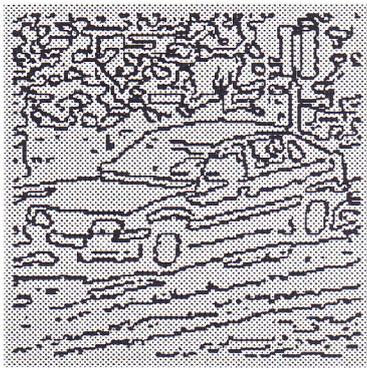
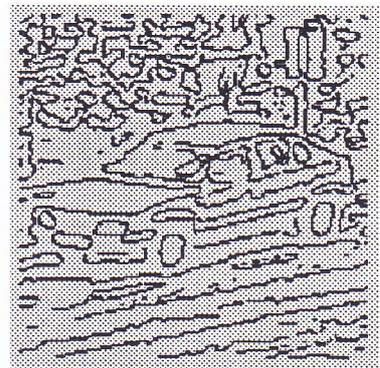


Figura D.7

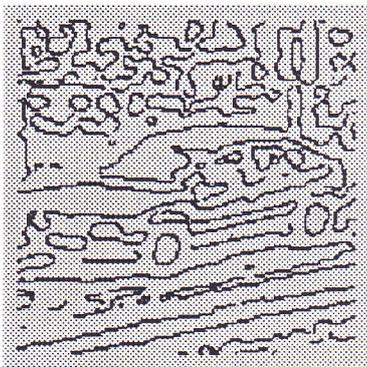
M6%2



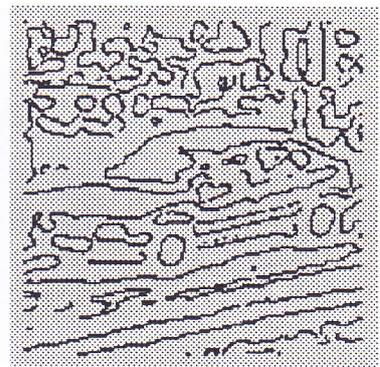
D1U601.8B4657k5K9



D2U163.2B4306k5K15



D2.5U459B3516k7K19



D3U433.5B3452k7K23

Figura D.8

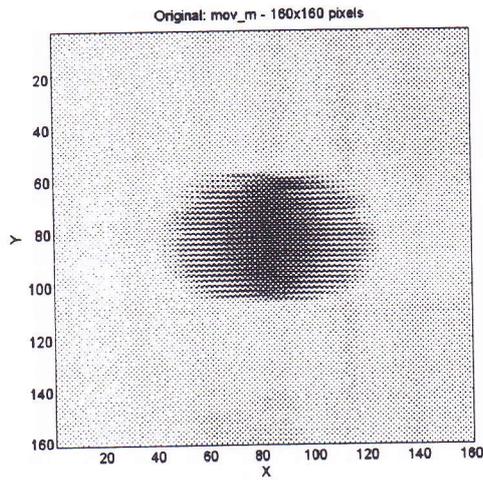
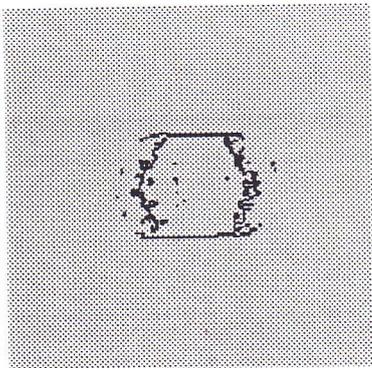
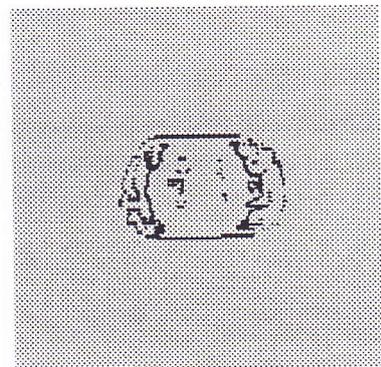


Figura D.9

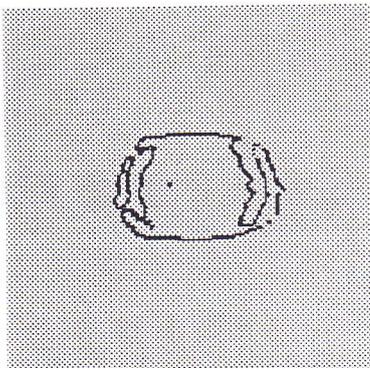
M6%2



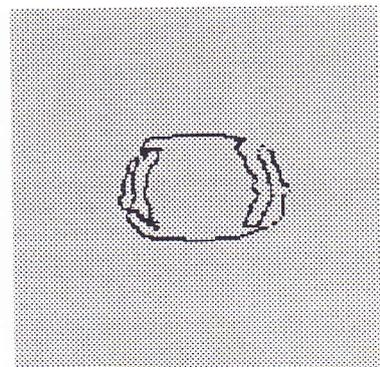
D1U601.8B363k5K9



D2U163.2B448k5K15



D2.5U459B392k7K19



D3U433.5B414k7K23

Figura D.10

En los ejemplos que siguen se presenta el resultado obtenido al aplicar el procedimiento propuesto a una imagen con mucha mayor definición que las anteriores. La imagen original empleada para esta segunda secuencia de resultados es la mostrada en la Figura 11.

Se puede observar que al aumentar la definición de 160x160 a 450x450, el comportamiento del filtro mejora de manera sustancial dado que cuenta con mayor información sobre las líneas que constituyen bordes en la imagen original.

Por otro lado, se puede observar la continuidad en el trazado de los bordes obtenidos, no obstante, la complejidad de las líneas que forman los bordes, y la variedad de tonos de grises presente en la misma.

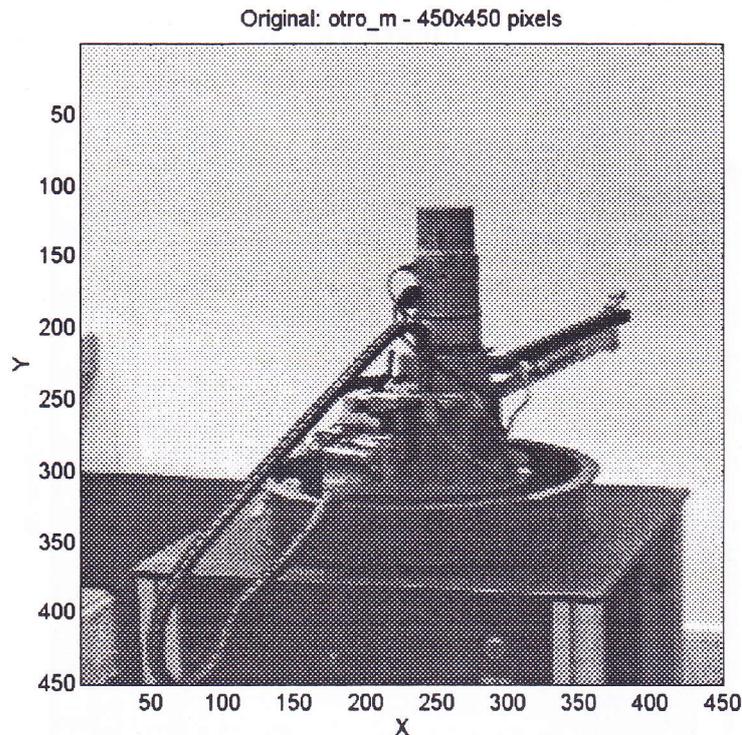


Figura 11

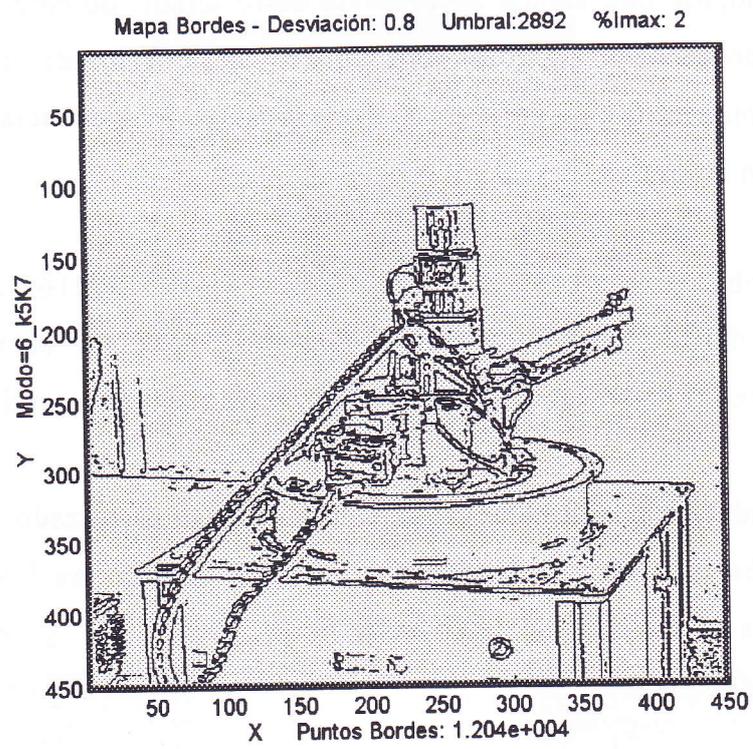


Figura 12

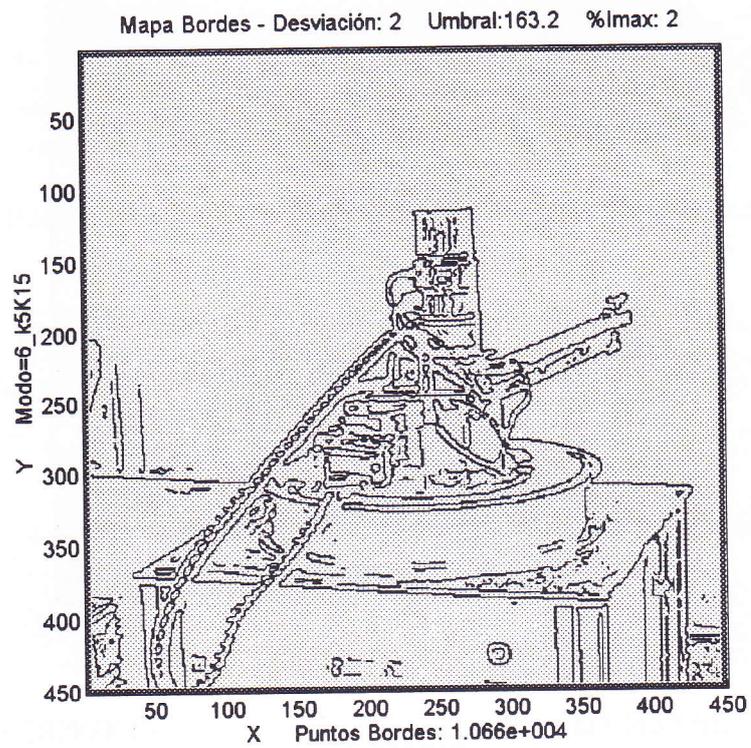


Figura 13

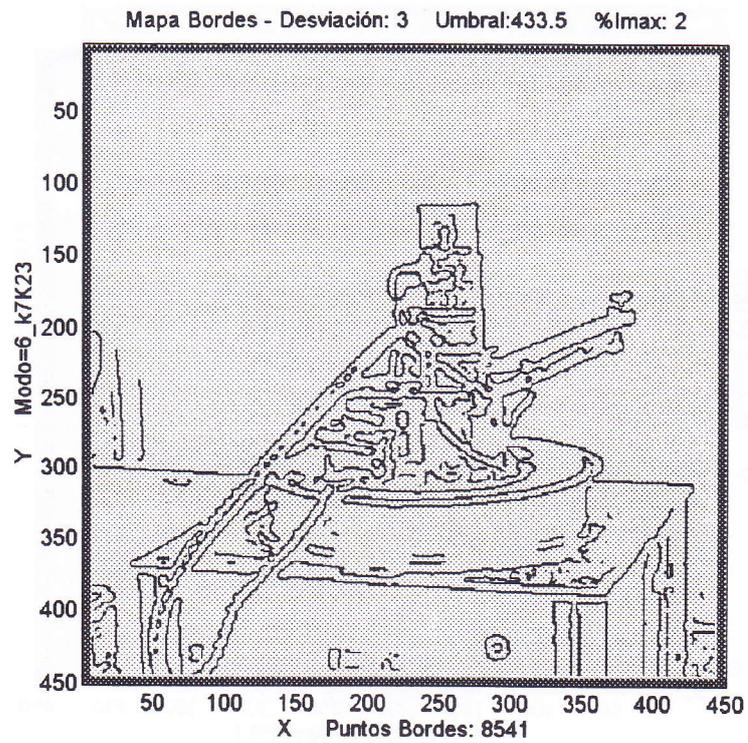


Figura 14

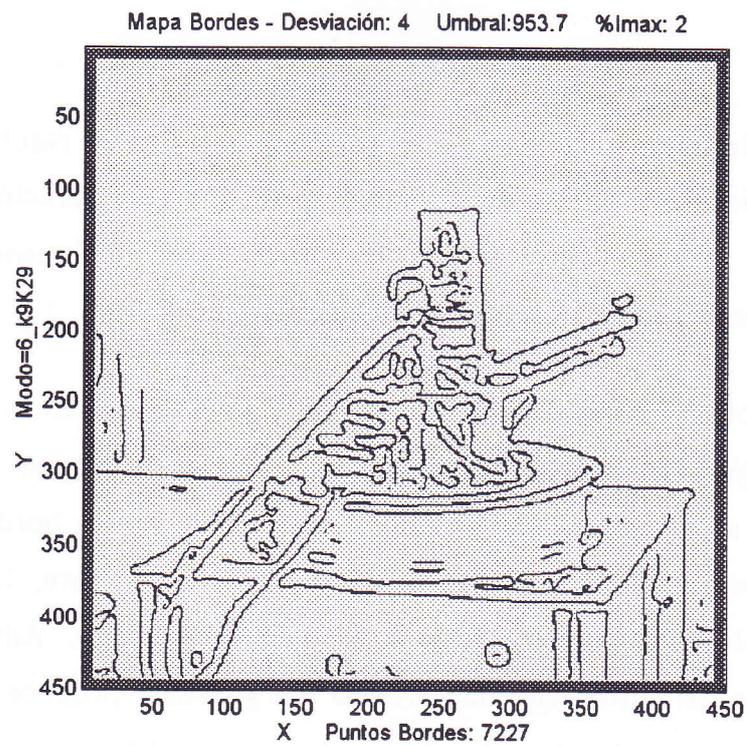


Figura 15

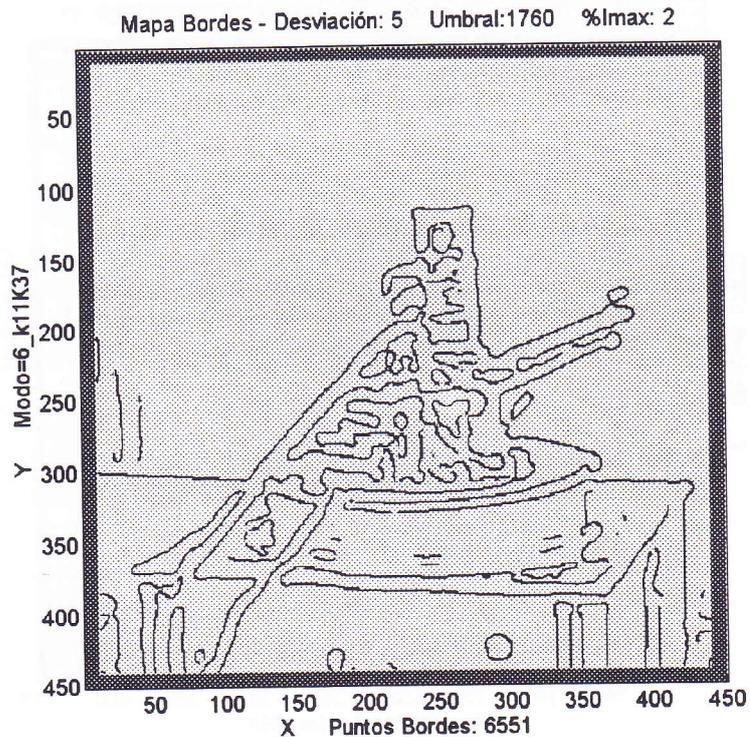


Figura 16

A modo de comparación se presenta a continuación el resultado obtenido para unas desviaciones típicas de 2 y de 5, para una configuración que trabaje obteniendo la segunda derivada de la imagen convolucionando la misma directamente con la segunda derivada del gaussiano.

Se puede observar que el resultado, cuando la desviación es 2, obtenido en la Figura 13 y Figura 14 es muy parecido. Sin embargo, al elevar el valor de la desviación de 2 a 5 (Figuras D.16 y D.18), el número de puntos borde encontrados disminuye drásticamente al cambiar de método. No obstante, la potencia de cálculo que se ahorra es muy grande para el método presentado. Adviértase que el excesivo tamaño del filtro segunda derivada del gaussiano produce la pérdida de gran cantidad de puntos borde en los límites de la Figura 18.

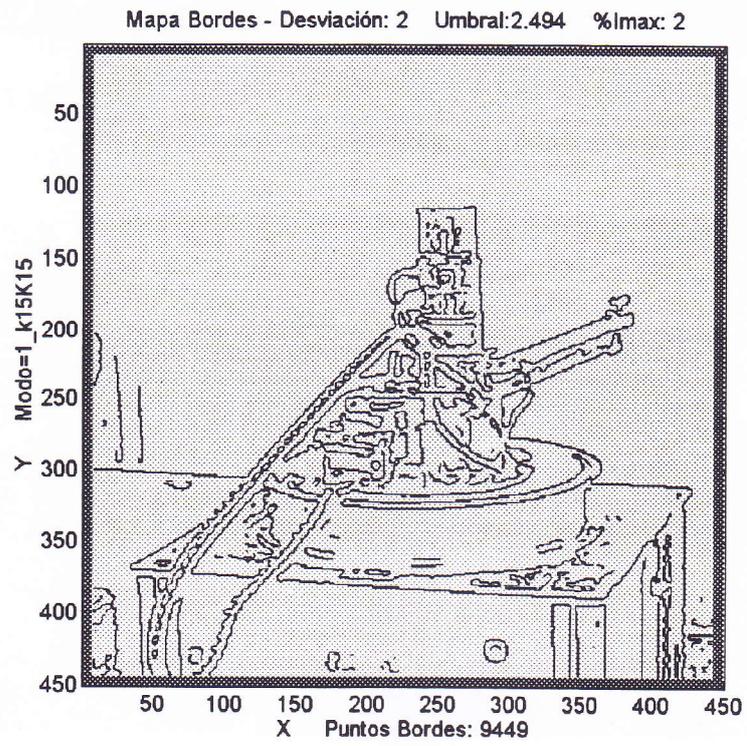


Figura 17

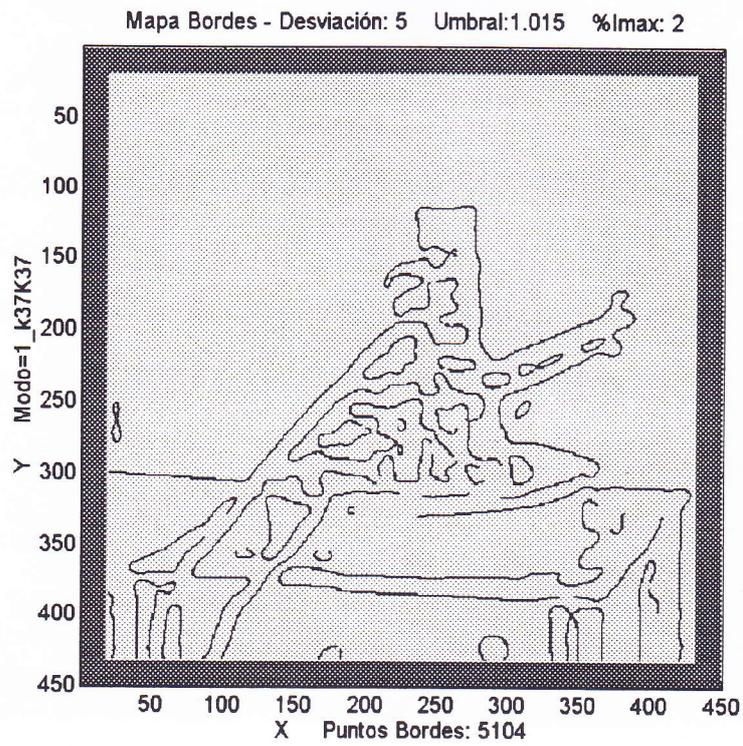


Figura 18

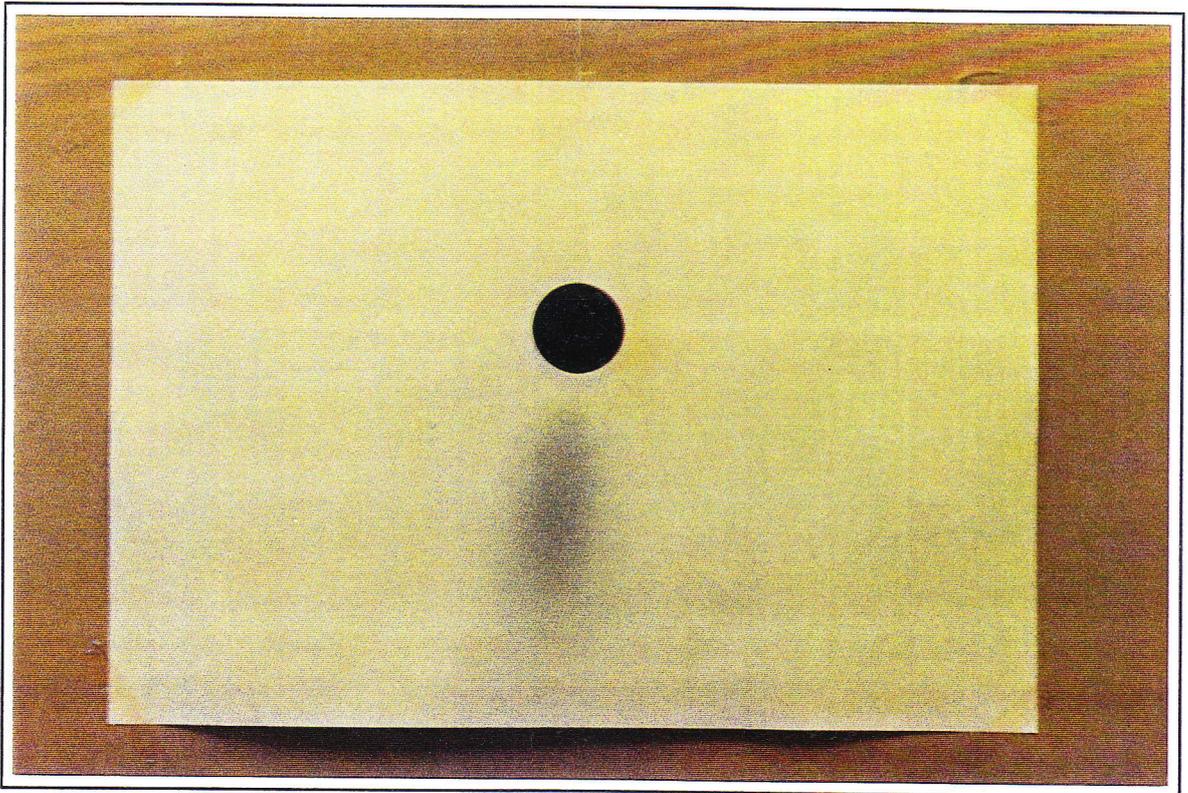


Ilustración D-1 Fotografía del péndulo.

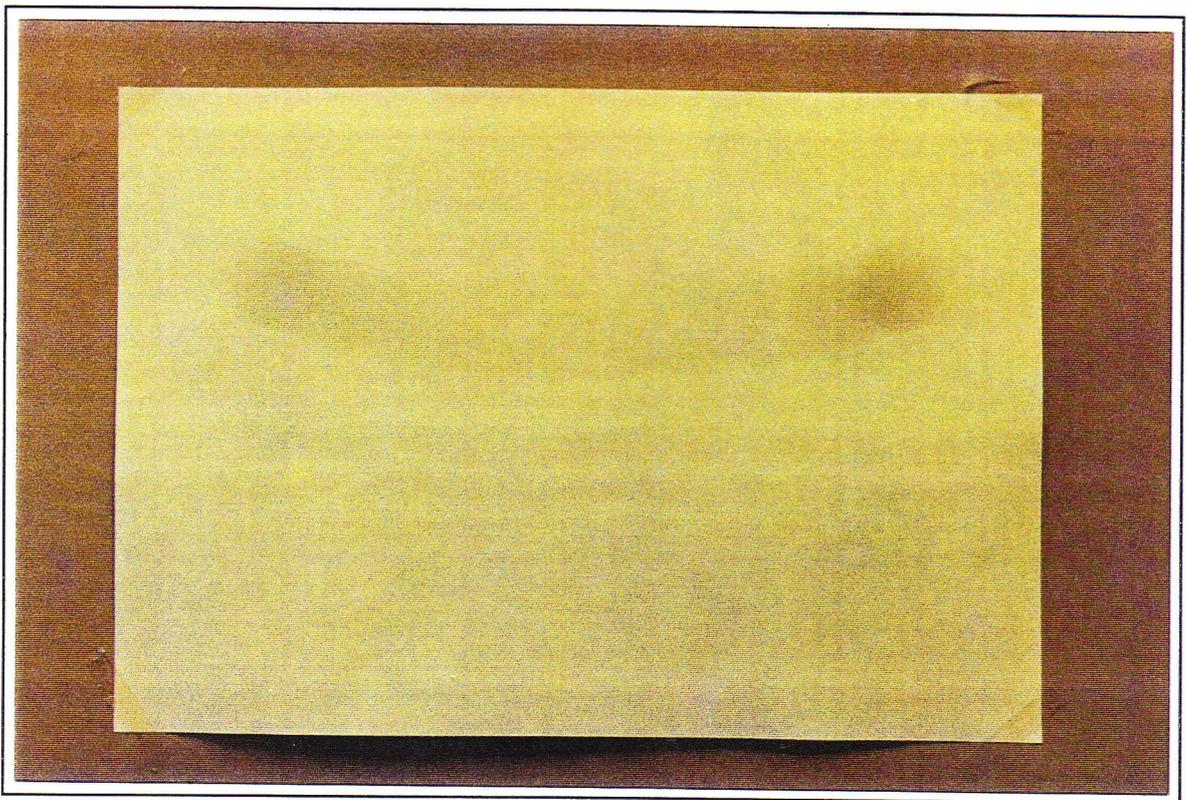


Ilustración D-2 Trayectoria seguida al moverse la bola.

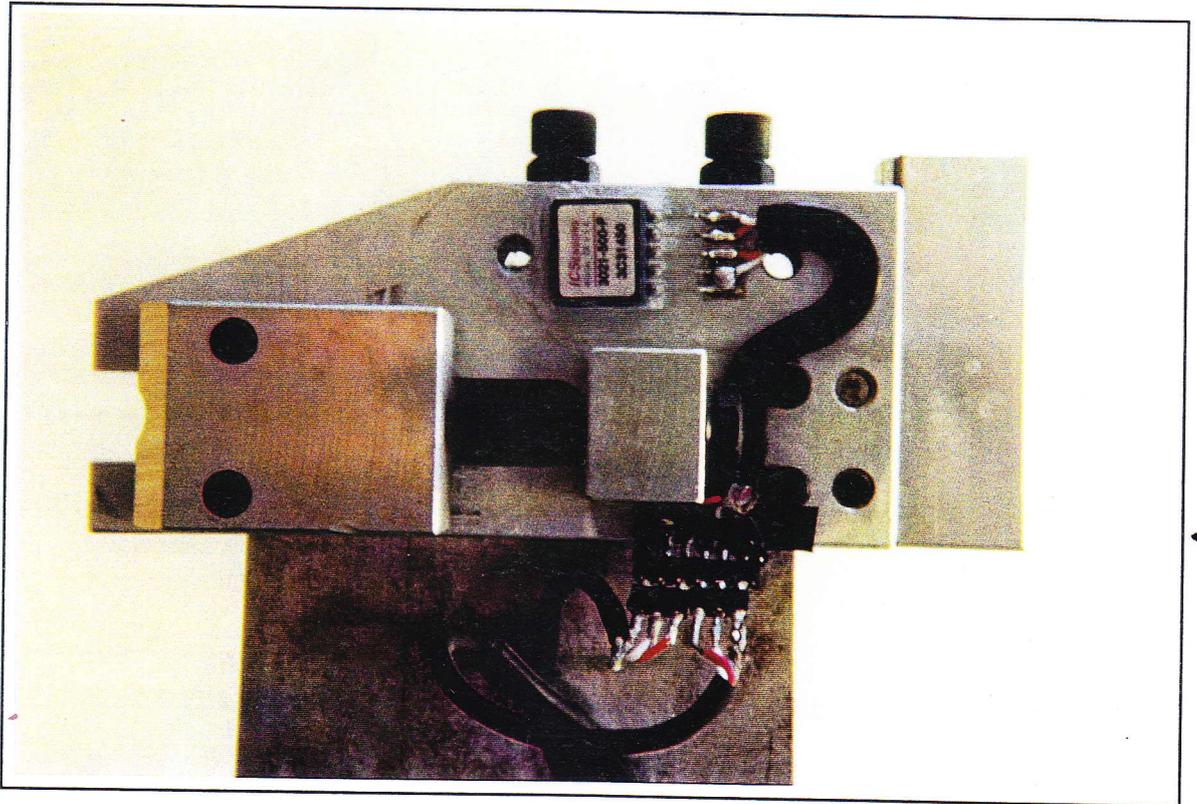


Ilustración D-3 Detalle del extremo del brazo flexible.

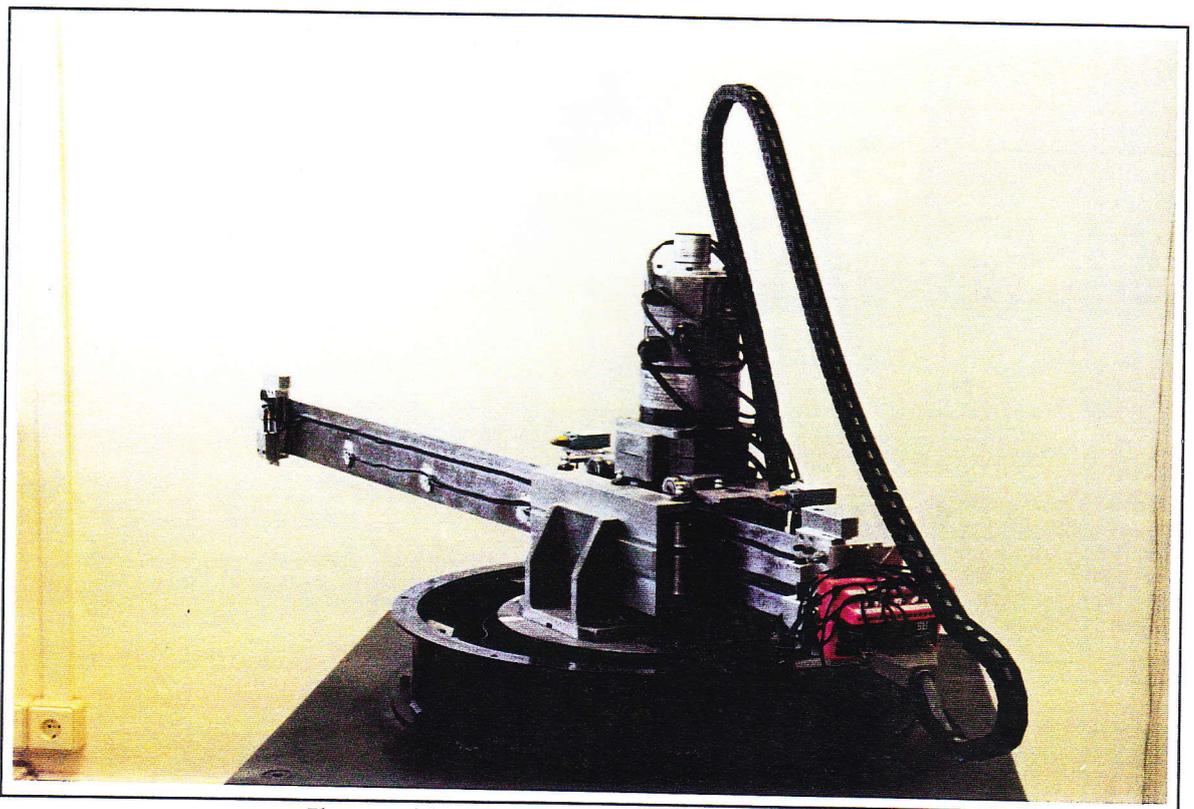


Ilustración D-4 Brazo de robot flexible.

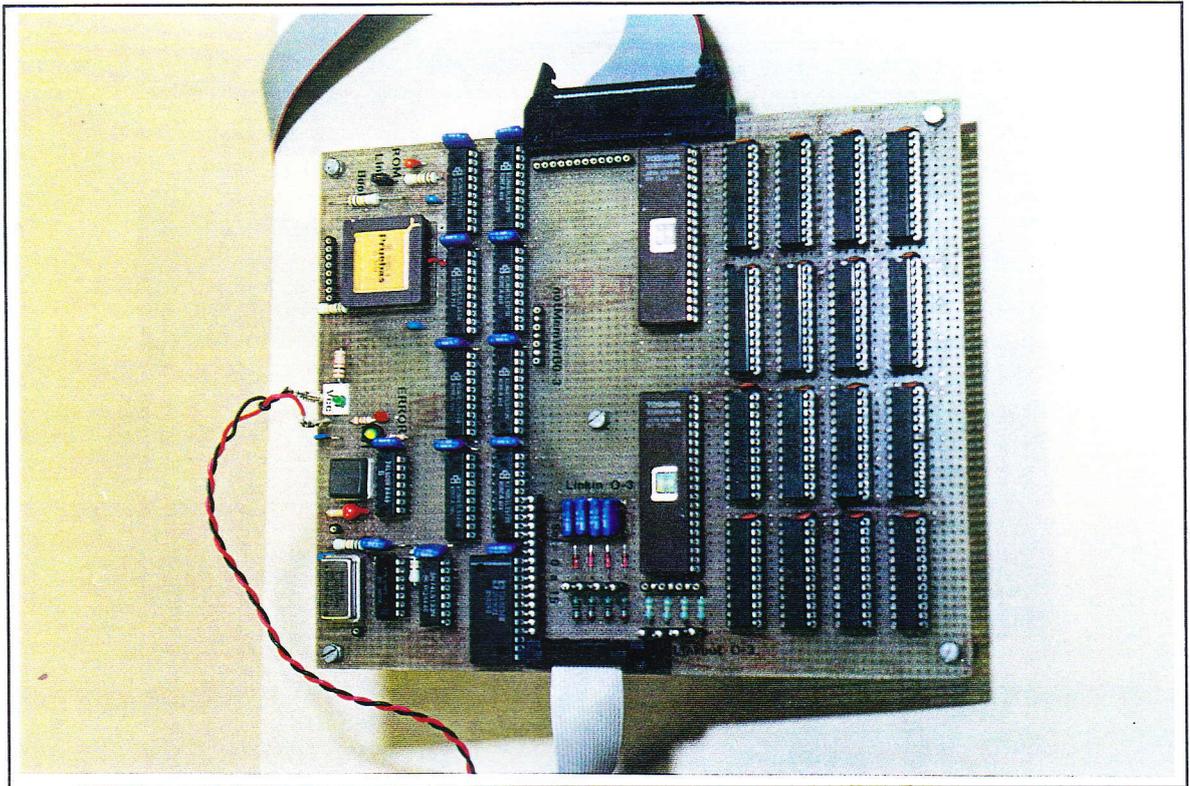


Ilustración D-5 Detalle de la Tarjeta que controla el DSP.



Ilustración D-6 Equipo Empleado.



Ilustración D-7 Cámara B/N utilizada.

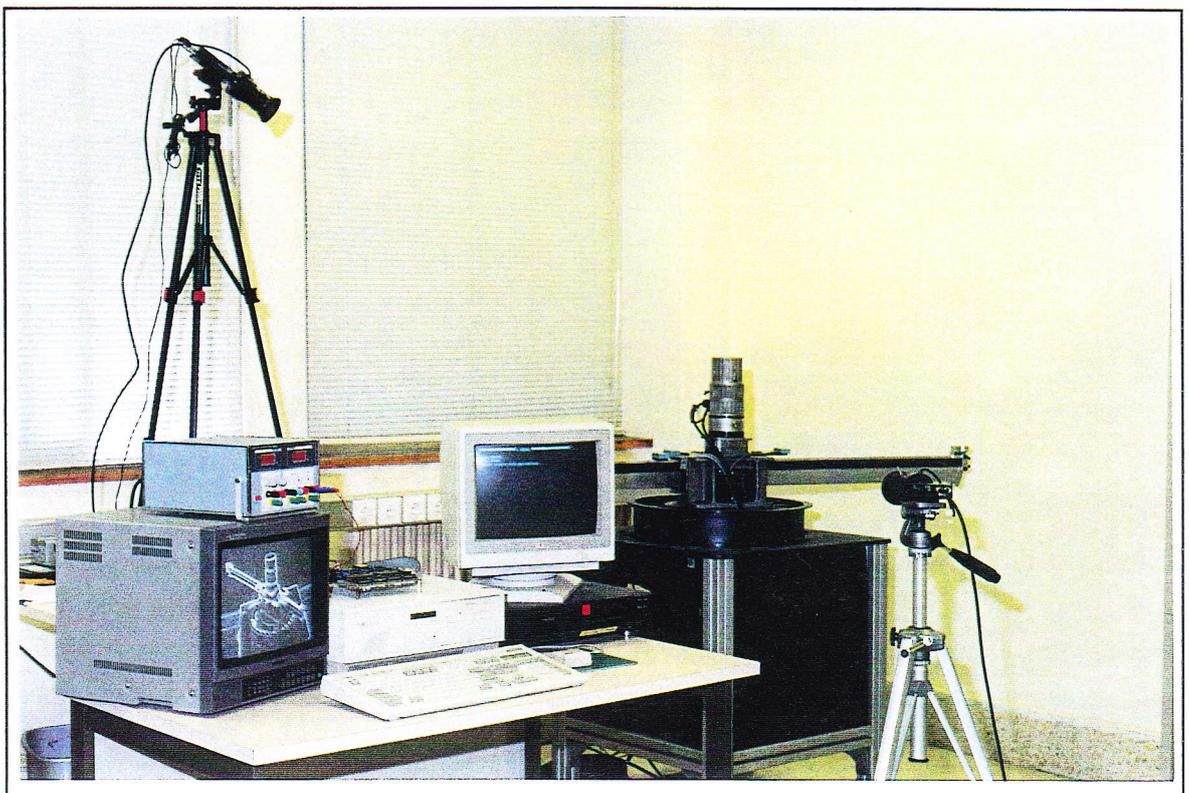


Ilustración D-8 Sistema de visión artificial.

