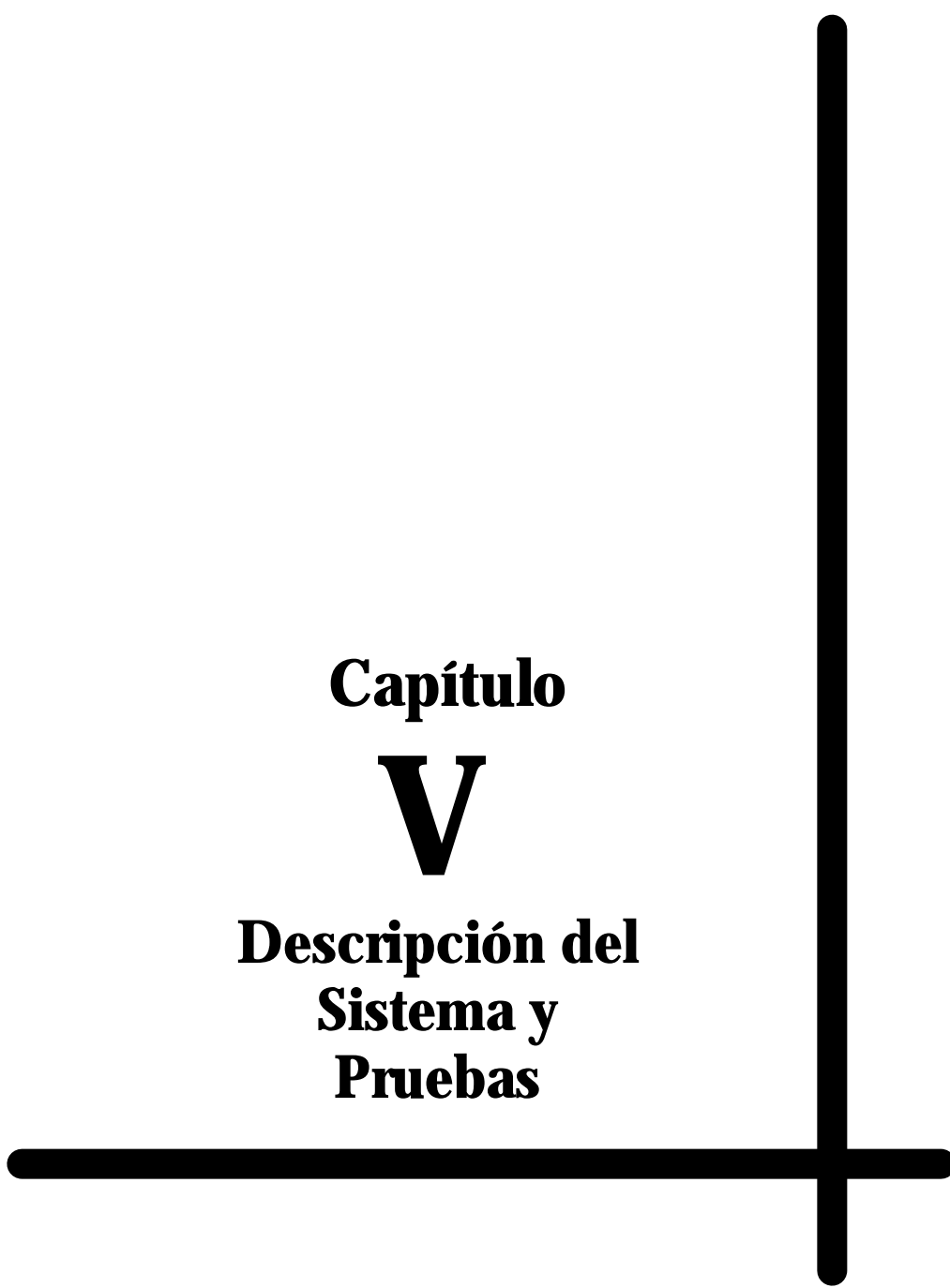


Capítulo

V

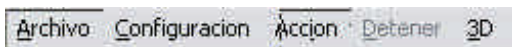
**Descripción del
Sistema y
Pruebas**



V

5.1 Descripción del Sistema

El sistema finalmente implantado es mostrado en las pantallas siguientes, se tiene una pantalla principal de aplicación, en la parte central superior se tiene una pequeña visualización del dispositivo de entrada en ese momento, se cuenta con seis lienzos donde se muestra el estado actual del procesado digital de las imágenes y de estos tres corresponden a la cámara izquierda (localizados en la parte izquierda de la aplicación) y tres a la cámara derecha (localizados en la parte derecha de la pantalla); de éstos, los primeros dos (uno para cada cámara) corresponden al estado de entrada, los siguientes presentan el estado del proceso después de haber realizado detección de borde y binarización; finalmente los dos inferiores presentan a las imágenes convertidas a niveles de gris con sus correspondientes vectores si se ha seleccionado presentarlos, en la parte central de la pantalla se encuentra un área de texto donde es posible visualizar las características obtenidas tanto de la imagen izquierda como de la derecha, así como su emparejamiento y finalmente el valor de la coordenada 3D obtenida.



En la parte superior de la pantalla principal se encuentra una barra de menú, y contiene 5 opciones, las cuales son *Archivo*, *Configuración*, *Acción*, *Detener* y *3D*.



Una barra de botones de Acceso rápido localizada debajo de la barra de menú, nos permite terminar el programa al presionar el primer botón, el segundo iniciar o detener la ejecución del sistema y el último botón nos permite el acceso de manera rápida a las características 3D.

Finalmente en la parte inferior se encuentra una barra de estado donde se advierte de las acciones más importantes del sistema.

La siguiente imagen es el modelo de la pantalla principal (ver fig. 5.1).

Se utilizaron formas estándares de construcción de pantallas para una fácil identificación y manejo de los elementos y dotándose de elementos de alto contraste para facilitar el uso.

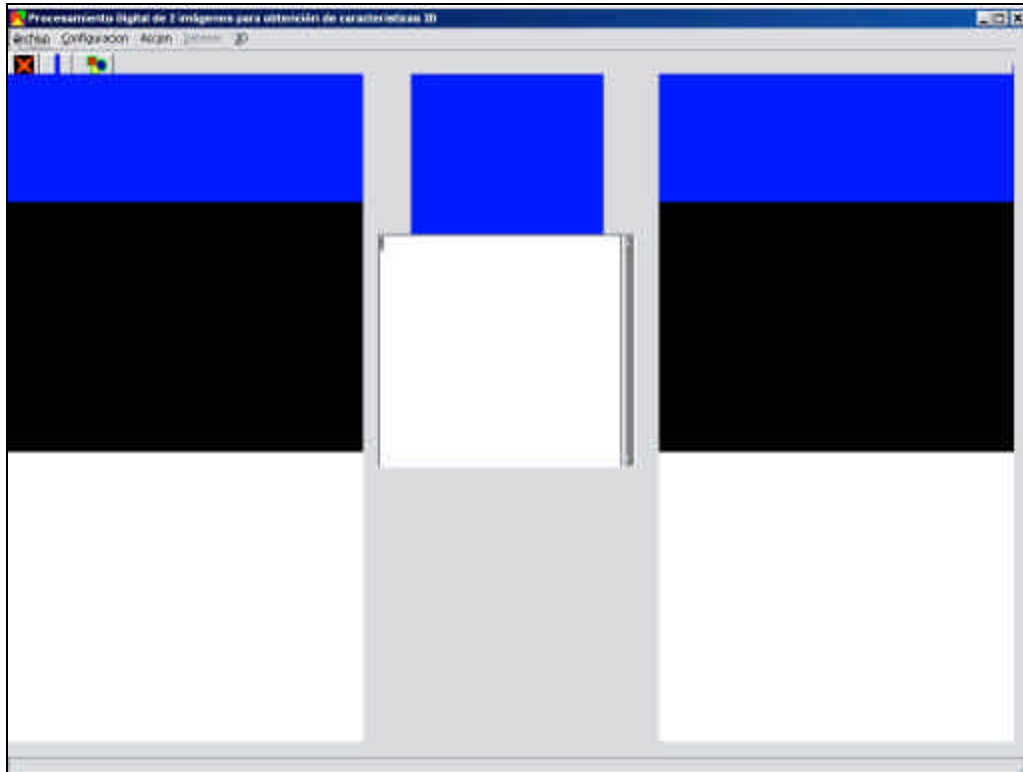


Figura 5.1 Pantalla Principal del sistema

La pantalla de calibración tanto de cámara como del color que se accesa desde la opción *calibración* del menú *configuración*, es mostrada en la siguiente figura. (ver fig. 5.2)

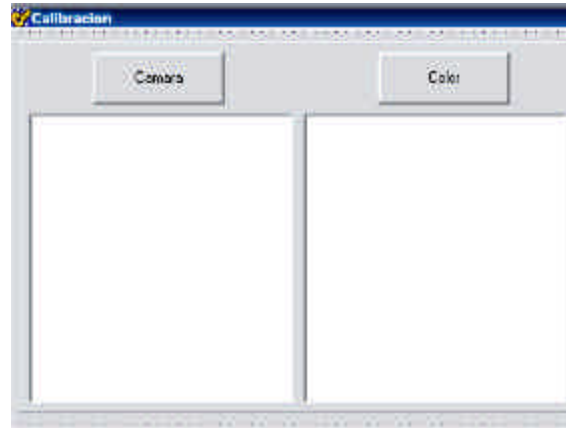


Figura 5.2 Pantalla Calibración

Como se había mencionado anteriormente en la opción *Configuración Threshold* dentro del menú *Configuraciones*, es posible realizar cambios a los umbrales de la aplicación, al ser seleccionada esta opción se abre la pantalla de configuraciones mostrada más adelante; aquí es donde es posible cambiar el umbral de binarización (parte superior izquierda); el umbral del ajuste de contraste sigmoidal (parte superior derecha); el umbral de las características necesarias para ser considerado un conjunto de características como vector (segunda opción a la izquierda); el número de zonas en que es dividido o el incremento de ángulos para búsqueda de vectores en la transformada hough (segunda opción a la derecha); el parámetro del espaciamiento existente entre las líneas que se escanean y comparan (parte inferior izquierda); configurar la distancia existente entre las cámaras (parte inferior derecha); y finalmente el umbral de color (parte inferior izquierda); la pantalla correspondiente a los umbrales es mostrada a continuación.

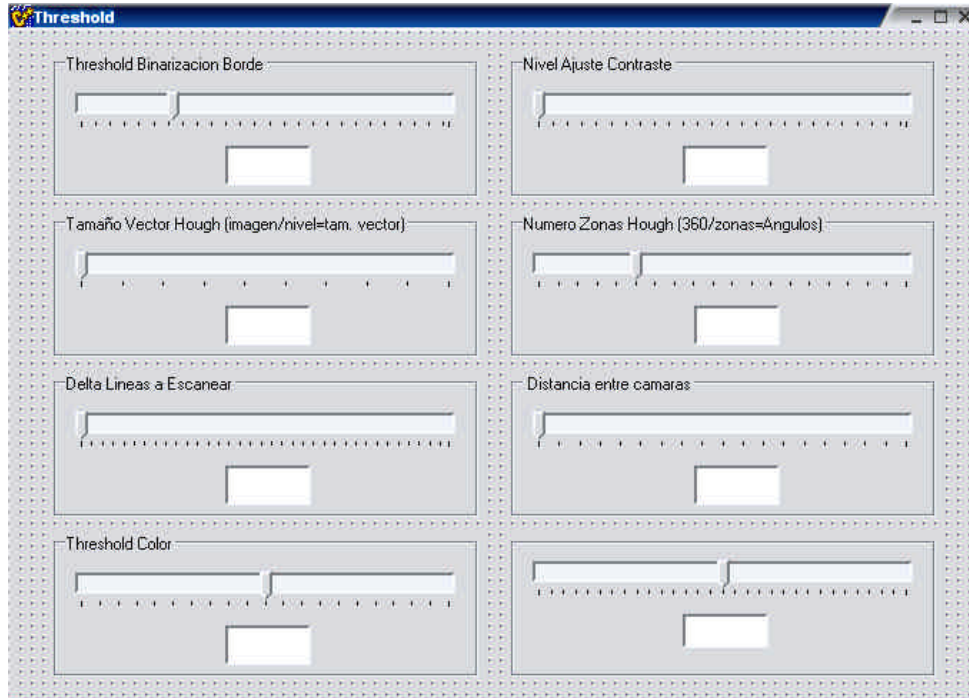


Figura 5.3 Pantalla de configuración

La pantalla final de representación tridimensional que es accesada desde el menú *Pantalla Open GL* del menú *3D*, es mostrada a continuación (ver fig. 5.4); tiene en su parte derecha los controles de control de visualización, dentro de los controles de visualización en la parte superior se encuentran los controles acerca de parámetros como visualización de la escena con luz (activando la opción *light*); el control de la visualización de los objetos suavizados o no (casilla de opción *Smooth*); la visualización de los objetos únicamente como estructuras mostrando sus aristas o bien como objetos sólidos (opción *Fill*) y la última opción es la visualización de la escena en 2D (opción *2D*).

En la parte inferior se encuentran 4 barras de desplazamiento, las primeras 3 corresponden a los controles de rotación sobre el eje X, Y y Z respectivamente y la última barra de desplazamiento sirve para indicar la distancia de visualización de una escena, es decir cambia la Z del observador.

Los últimos controles que se encuentran en esta pantalla son los de impresión y guardar que se encuentran en la parte superior izquierda del área de control y sirven para salvar la escena generada en un archivo gráfico o enviar la misma a impresión.

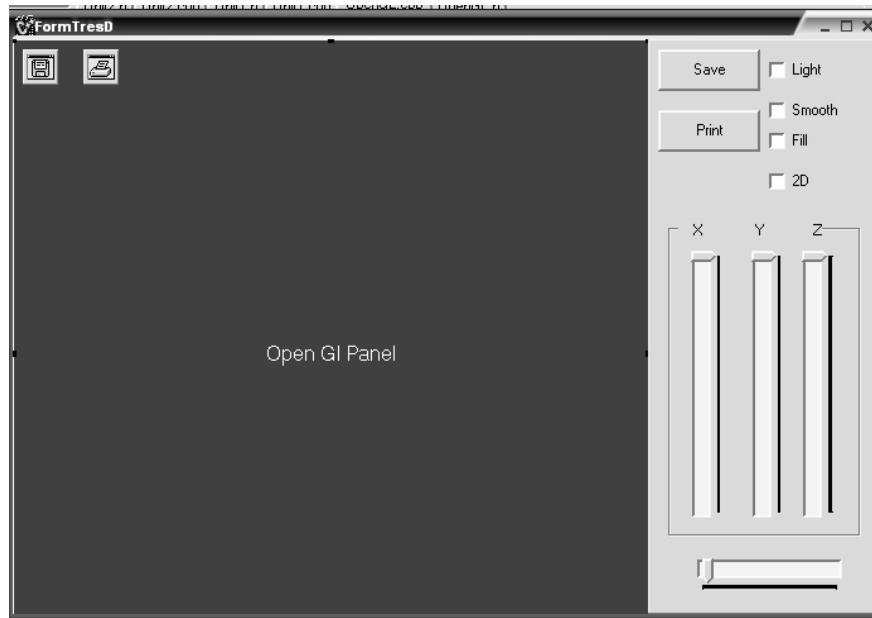


Figura 5.4 Pantalla 3D

Físicamente el sistema es como se muestra a continuación (ver Fig. 5.4b)



Fig. 5.4b Sistema de cámaras

5.2 Pruebas

Se hicieron pruebas del procesamiento digital de imágenes en Java, Visual Basic y diferentes versiones de C++ resultando el mejor para este propósito el compilador C++ de Borland y puesto que en este trabajo el tiempo de ejecución es un factor crucial, resultó éste último ser una muy buena herramienta, además de que el tiempo de desarrollo de la interfaz es reducido debido a la orientación visual del mismo, por lo que se puede concentrar en los aspectos de código desde el inicio.

Para encontrar el sistema operativo que mejor funcione a estas necesidades, fueron programadas rutinas de multiplicaciones y divisiones de números de punto flotante así como sumas de enteros, encontrándose que el mismo programa corriendo bajo XP es considerablemente más rápido que corriendo bajo Windows 98/95 y es casi igual de rápido que corriendo en un sistema con Windows 2000.

Las pruebas de la corrida del programa se hicieron en un S.O. Windows 98 y Windows XP, sin embargo, el desempeño del sistema fue notablemente mejor en el sistema operativo Windows XP.

El tiempo empleado para ejecutar operaciones de punto flotante y enteras, es insignificamente menor para las de tipo entero, por lo que no se justifica el programar algoritmos especiales que hagan uso de este tipo de operaciones.

Las imágenes bajo algunas condiciones no eran procesadas de manera correcta, es decir, había cierta confusión entre fondo y figura(s) por lo que se probó aplicar un algoritmo que diera cierto contraste y permitiera discernir mas fácilmente, para esto fue probado el uso de un algoritmo de contraste de tipo sigmoidal, al aplicar este algoritmo, la separación entre fondo y figura se mejora en gran medida, además nuestra implantación permite modificar la cantidad de contraste que se necesita aplicar; una de las pruebas que resultaron en una mayor confusión fue el uso de un cilindro amarillo, en la figura 5.5 se muestra la imagen de prueba sin y con la aplicación de un contraste sigmoidal, en este caso el parámetro para el contraste fue fijado en 120 en un rango de 0 a 255.

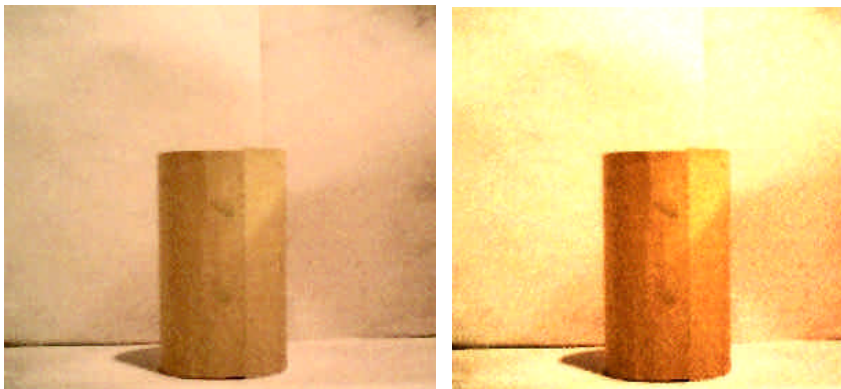


Fig. 5.5 Resultados de aplicar un contraste

Se hicieron pruebas usando distintos tipos de filtrado de suavizamiento como es el caso del filtrado pasabajas y el filtrado gaussiano, además de usar este filtrado, fue probado el no usar ningún tipo de filtrado antes de la detección de borde.

Se encontró que una de las mejores combinaciones, es el uso de realce sigmoidal medio (parámetro igual a 128 de 255) junto con cualquiera de los filtrados ya sea gaussiano o pasabajas.

Un correcto umbral de binarización también contribuye en gran medida a la mejor detección de características que puedan servir para el emparejamiento estéreo, si el umbral es alto únicamente cambios grandes en los bordes son detectados; si es bajo, cambios apenas perceptibles pueden ser detectados. En general se hace uso de un umbral de 50 para la binarización en ambos detectores de bordes, pero en muchas ocasiones cuando el número de características detectadas es bajo se usa un umbral de 32 y aun menos si así es requerido.

Después de aplicar los algoritmos de detección de bordes Roberts, Sobel, Prewitt y Canny a distintas imágenes y secuencias de imágenes provenientes de cámaras, finalmente se escogieron dos algoritmos por ser los que mejores resultados presentaron. La eficiencia de su resultado estuvo determinada por la exactitud en el reconocimiento de bordes, estos algoritmos fueron Sobel y Canny, que además de ser los más representativos son entre los de mayor uso. Otros tipos de detectores de bordes como el Prewitt, es muy parecido al Sobel y los resultados obtenidos de igual manera son similares.



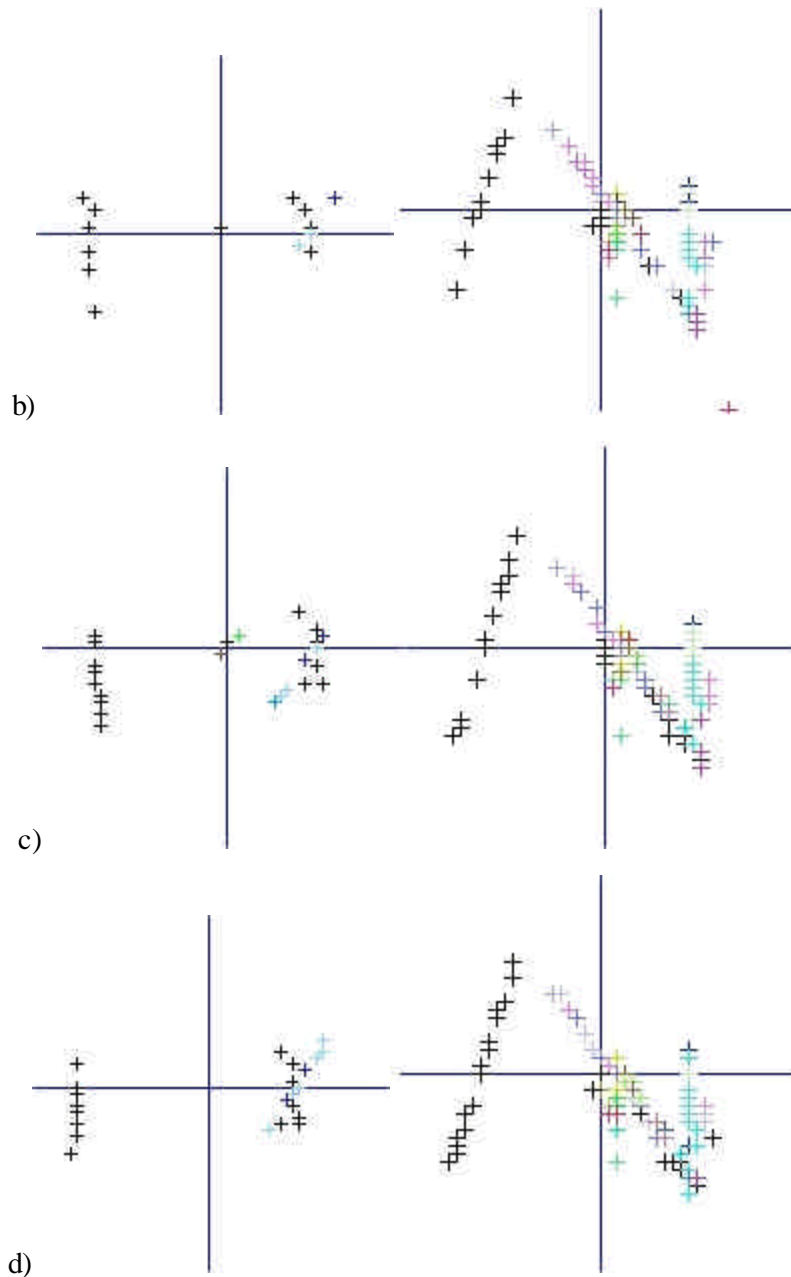


Fig. 5.6 a) Escena analizada b) Emparejamiento con pixel central c) Emparejamiento sin pixel central
d) Emparejamiento sin pixel central y con inclinación

Para obtener la diferencia entre pixeles al aplicar los algoritmos de emparejamiento estéreo, se probaron varios métodos; el primero que se realizó fue el emparejamiento simple, es decir; la simple comparación del nivel de gris del pixel de la característica de la imagen izquierda con el pixel de la imagen derecha; esta forma resultó ser muy poco efectiva, el siguiente fue el evaluar los valores izquierdo, central y derecho alrededor de la característica de interés de la imagen a escala de gris

izquierda contra los píxeles izquierdo central y derecho de la imagen derecha, esta segunda aproximación resultó ser un poco mejor, (ver fig. 5.6 b) aunque distaba mucho de los resultados deseados; una tercera forma de evaluación involucró la evaluación de los píxeles izquierdo y derecho sin considerar el central, con esto se obtuvieron buenos resultados (ver fig. 5.6c); sin embargo los mejores se obtuvieron cuando además se añadió la comparación de la inclinación del borde, es decir la consideración del vector característico (ver fig. 5.6 d). Los resultados obtenidos mejoraron bastante para los dos algoritmos de detección de borde, cabe destacar que aunque la mejora fue mayor para el algoritmo Sobel no significó llegar a alcanzar los buenos resultados obtenidos utilizando el algoritmo Canny.

Como prueba inicialmente el matching se realizó únicamente con las características obtenidas de la línea epipolar de las imágenes, como puede observarse en la figura 5.7 el matching se está realizando únicamente sobre una línea, en este caso es la línea epipolar, es decir línea 144, en la imagen izquierda se determinan 3 características a emparejar (59,151,244) contra 3 características de la imagen izquierda (76,120,184); en este caso se obtienen 3 pares conjugados (59,76 / 151,120 / 244,184). Otra prueba realizada de matching es en este caso a un cubo (ver fig. 5.8), en este caso sobre la línea epipolar se obtienen 2 características (16,267) para la imagen izquierda y (46,295) para la derecha, después del emparejamiento se obtienen los pares (16,46 / 267,295).

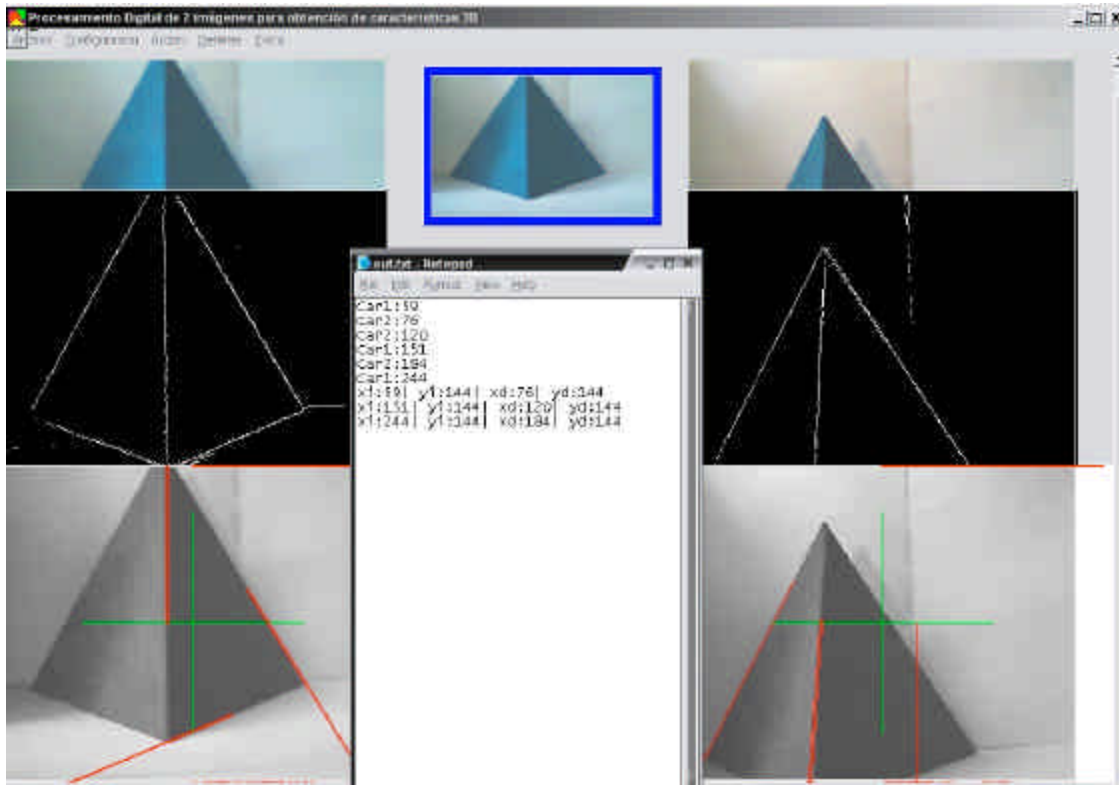


Fig 5.7 Prueba matching triangulo

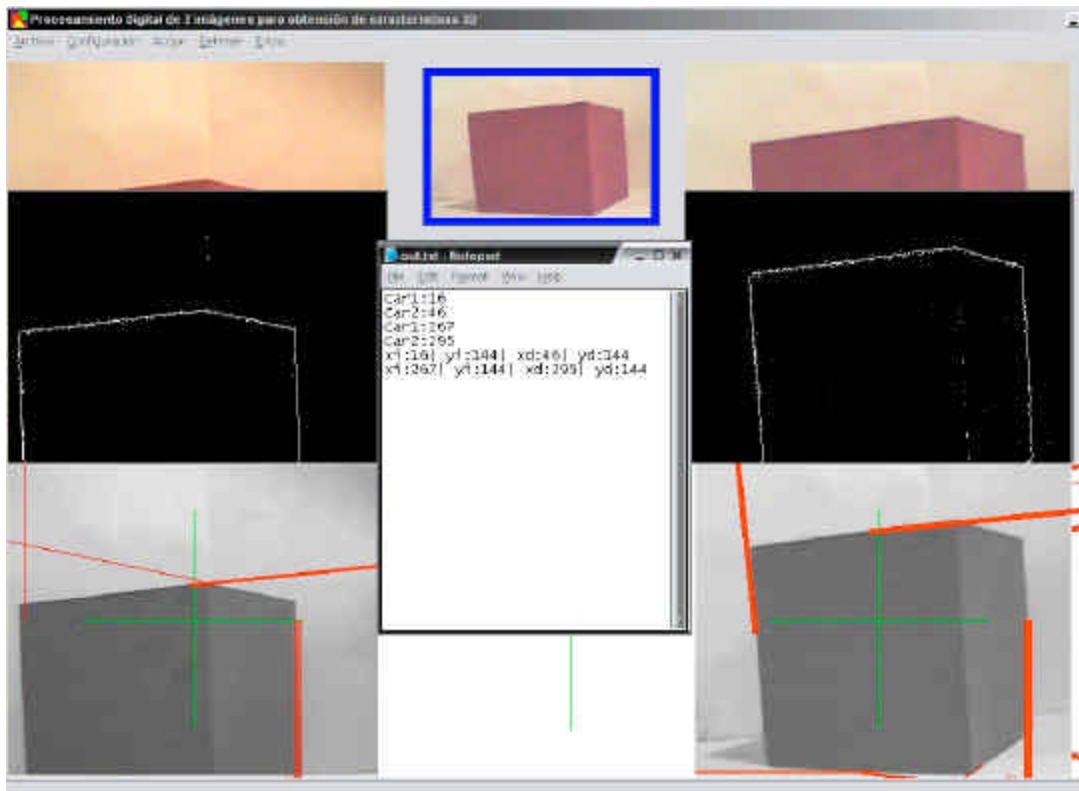


Fig. 5.8 Prueba matching cubo

El resultado fue satisfactorio, por lo que se procedió a realizar la corrida del programa haciendo el barrido de las características sobre varias líneas de la imagen (aproximadamente cada 25 líneas, aunque el parámetro puede ser modificado) y también se realizaron pruebas haciendo barrido cada 10 y cada 8 líneas.

Para obtener la coordenada Z se analizó el uso de un ajuste de tipo exponencial, de igual manera se implantó el uso de un polinomio de quinto grado.

El ajuste exponencial presentó algunos errores, por lo que se procedió a encontrar un polinomio de quinto grado para obtener la coordenada Z , sin embargo, al realizar la extrapolación, los valores suelen tener un comportamiento muy errático especialmente para una disparidad grande en los pixeles, por lo que se obtuvo un polinomio de quinto grado con los valores más extremos posibles para que la extrapolación en caso de ser necesaria sea mínima y sólo en casos muy especiales. Así que de esta manera al eliminar datos intermedios y no extremos, se hace que la extrapolación se acorte en los límites; incluso llegando a tener mejores valores que con el ajuste tipo exponencial.

El rango de error para el ajuste exponencial es en promedio entre 7.15% es decir: 4.6 cms. Cuando la separación de las cámaras es de 14 cms. Y hasta el 16.1% 9.4 cms. Cuando la separación de las cámaras es de 7cms. Para el resultado por polinomio exponencial es en promedio entre 2.6% es decir: 1.6 cms. Cuando la separación de las cámaras es de 7 cms. Y hasta el 11.3% 12.5 cms. Cuando la separación de las cámaras es de 10 cms; para la obtención de la distancia por medio de fórmula es en promedio 1.0% es decir: 1.0 cms. cuando la separación de las cámaras es de 14 cms. y máximo el 2.9% 2.0 cms. cuando la separación de las cámaras es de 10 cms.

Los resultados de este tipo de ajustes son mostrados en la Tabla 5.1 en donde se tienen los 10 datos obtenidos experimentalmente, los resultados obtenidos por medio de fórmula, aproximación del polinomio de quinto grado y por medio del ajuste exponencial. La comparación mediante los dos métodos es mostrada en la Tabla 5.1.

Distancia Pixeles	Distancia Real (cm)	Formula	Ajuste Exponencial	Polinomio grado 5
306	14.5	14.87	11.7	14.5
193	23	23.58	27.3	23.0
110	43	41.36	50.8	38.7
101	45	45.05	54.3	45.0
84	51	54.17	61.7	55.1
73	62	62.33	67.0	62.0
59	75	77.12	74.5	75.7
51	89	89.22	79.1	89.0
44	99	103.41	83.3	106.0
42	112	108.33	84.6	112.0

a

Formula		Exponencial		Polinomial	
Error cms.	Error %	Error cms.	Error %	Error cms.	Error %
0.4	2.48	2.8	24.12	0.0	0.00
0.6	2.44	4.3	15.63	0.0	0.00
1.6	3.96	7.8	15.35	4.3	11.19
0.0	0.11	9.3	17.20	0.0	0.00
3.2	5.85	10.7	17.39	4.1	7.47
0.3	0.53	5.0	7.52	0.0	0.00
2.1	2.75	0.5	0.72	0.7	0.95
0.2	0.24	9.9	12.56	0.0	0.00
4.4	4.26	15.7	18.80	7.0	6.58
3.7	3.38	27.4	32.40	0.0	0.00
1.7	2.60	9.4	16.169929	1.6	2.6190708

Distancia Pixeles	Distancia Real (cm)	Formula	Ajuste Exponencial	Polinomio grado 5
309	21.5	22.65	18.91	21.5
239	28.5	29.29	29.99	140.9
194	35.5	36.08	40.35	35.5
150	46	46.67	53.92	35.3
124	56	56.45	64.00	56.0
106	66.5	66.04	72.06	68.8
94	77	74.47	77.99	77.0
84	85.5	83.33	83.30	85.6
75	97	93.33	88.39	97.0
68	110	102.94	92.56	110.0

b

Formula		Exponencial		Polinomial	
Error cms.	Error %	Error cms.	Error %	Error cms.	Error %
1.2	5.09	2.6	13.70	0.0	0.00
0.8	2.69	1.5	4.98	112.4	79.77
0.6	1.61	4.8	12.01	0.0	0.00
0.7	1.43	7.9	14.69	10.7	30.23
0.5	0.80	8.0	12.49	0.0	0.00
0.5	0.70	5.6	7.71	2.3	3.33
2.5	3.40	1.0	1.26	0.0	0.00
2.2	2.60	2.2	2.64	0.1	0.14
3.7	3.93	8.6	9.74	0.0	0.00
7.1	6.86	17.4	18.84	0.0	0.00
2.0	2.91	6.0	9.807624	12.5	11.346766

Distancia Pixeles	Distancia Real (cm)	Formula	Ajuste Exponencial	Polinomio grado 5
316	27.5	27.47	24.27	27.5
241	36.5	36.02	37.50	27.7
190	46	45.68	50.40	46.0
163	53	53.25	58.95	54.0
140	62	62.00	67.36	62.0
119	73	72.94	76.09	72.3
104	82.5	83.46	83.00	82.5
94	91	92.34	87.96	91.0
84	101	103.33	93.21	101.0
77	109	112.73	97.08	109.0

c

Formula		Exponencial		Polinomial	
Error cms.	Error %	Error cms.	Error %	Error cms.	Error %
0.0	0.12	3.2	13.32	0.0	0.00
0.5	1.34	1.0	2.65	8.8	31.54
0.3	0.69	4.4	8.73	0.0	0.00
0.3	0.47	5.9	10.09	1.0	1.87
0.0	0.00	5.4	7.96	0.0	0.00
0.1	0.08	3.1	4.06	0.7	0.97
1.0	1.15	0.5	0.61	0.0	0.00
1.3	1.45	3.0	3.46	0.0	0.02
2.3	2.26	7.8	8.35	0.0	0.00
3.7	3.31	11.9	12.28	0.0	0.00
1.0	1.09	4.6	7.1508805	1.0	3.4400931

Tabla 5.1 Distancias real, por fórmula, con ajustes exponencial y por polinomio

a) Distancia de las cámaras 7cms b)10cms c)14cms

Finalmente el programa tiene la posibilidad de obtener la coordenada Z mediante los tres métodos descritos anteriormente, sin embargo es por medio de fórmula que se obtienen los mejores resultados, las coordenadas X y Y, son obtenidas de igual manera mediante ecuación

Para probar la exactitud de los algoritmos, se realizaron pruebas localizando las cámaras a una distancia entre sí de 7, 10 y 14 centímetros respectivamente. Los resultados experimentales de la comparación de la disparidad en pixeles (que es inversamente proporcional a la distancia) contra la distancia real son mostrados en la figura 5.9 a, b y c.

Distancia Pixeles	Distancia Real (cm)
306	14.5
193	23
110	43
101	45
84	51
73	62
59	75
51	89
44	99
42	112

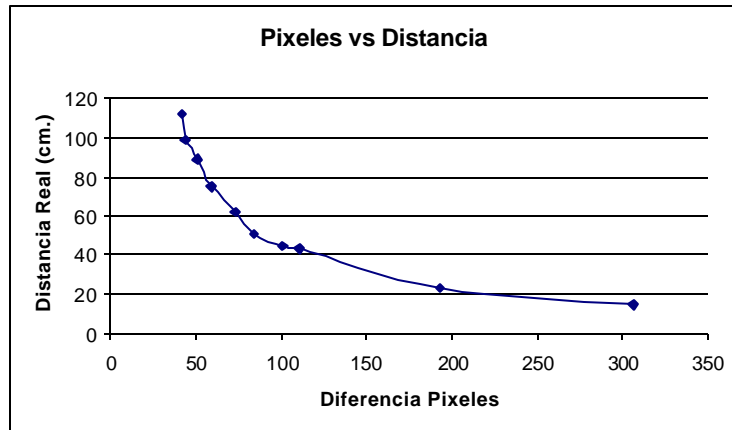


Fig. 5.9a Distancia en Pixeles Vs Distancia real Obtenidos Experimentalmente D=7cms

Distancia Pixeles	Distancia Real (cm)
309	21.5
239	28.5
194	35.5
150	46
124	56
106	66.5
94	77
84	85.5
75	97
68	110

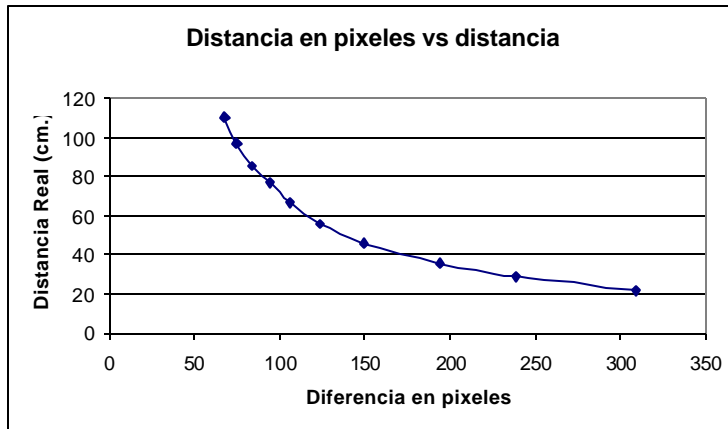


Fig. 5.9b Distancia en Pixeles Vs Distancia real Obtenidos Experimentalmente D=10cms

Distancia Pixeles	Distancia Real (cm)
317	27.5
241	36.5
190	46
163	53
140	62
119	73
104	82.5
94	91
84	101
77	109

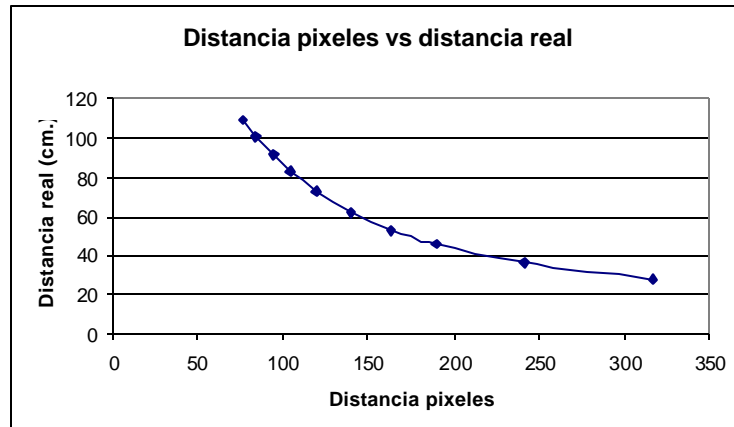


Fig. 5.9c Distancia en Pixeles Vs Distancia real Obtenidos Experimentalmente D=14cms

Como se puede apreciar entre mayor es la separación de las cámaras al procesar objetos que se encuentran a una distancia más lejana, son mejor detectados y procesados debido a que se trabaja en una región más lineal al haber una diferencia en píxeles mayor, por lo que se evita trabajar en la región asintótica que se presenta cuando la distancia del objeto es lejana y la separación de las cámaras es corta. Eso ayuda para la determinación con más precisión de la profundidad.

Para analizar el comportamiento general del sistema se montaron las cámaras sobre una regla de medición; se construyeron objetos geométricos de diversos colores y formas y se colocaron éstos a distancias que variaron entre los 30 centímetros y 1.10 metros sobre un fondo blanco; la iluminación fue proporcionada por una luz dicrónica de 60W colocada justo encima de las cámaras para mejor iluminación de la escena.