

Técnicas de Implementación de Visión Estereoscópica en Robótica

Cristian Daniel Ortega, Fernando Ernesto Moyano¹

¹Universidad del Norte Santo Tomás de Aquino - Facultad de Ingeniería.
Tucumán, Argentina
cristianortega86@gmail.com, ef.moyano@gmail.com

Coordinadores

Gustavo Raúl Sbrugnera, Tejerina José Gabriel Tejerina²
²Universidad del Norte Santo Tomás de Aquino - Facultad de Ingeniería.
Tucumán, Argentina

Técnicas de Implementación de Visión Estereoscópica en Robótica

Abstract: Se presenta el trabajo realizado como tesis final de grado de la carrera de Ingeniería Informática. Se investigó sobre la estereoscopia digital aplicada a la robótica con procesos en tiempo real. Capturando dos imágenes simultáneamente desde cámaras USB se pudo construir un modelo de profundidad a partir de la geometría del sistema estereoscópico propuesto mediante métodos de correspondencia de puntos homólogos.

Palabras claves: robótica, estereoscopia digital, correspondencia estereoscópica, modelo de profundidad, adquisición de imágenes.

1. Introducción

En los últimos años ha crecido sustancialmente el desarrollo de aplicaciones enfocadas a la tecnología de Visión por Computador. Uno de los conceptos que caracteriza a esta tecnología es la tridimensionalidad; en los robots, la sensación tridimensional se consigue mediante la obtención de imágenes estereoscópicas generadas por dos cámaras desplazadas, las cuales adquieren dos tomas distintas de la misma escena. La visión estereoscópica artificial se basa en el modelo estereoscópico biológico, donde el desplazamiento relativo de los ojos permite obtener la profundidad de los objetos o tercera dimensión mediante un simple proceso de triangulación a partir de las dos imágenes generadas por el mismo objeto de la escena 3D en cada ojo.

La finalidad básica del proyecto consiste en extraer el mapa de disparidad para obtener así la profundidad, teniendo en cuenta que la geometría de la cámara y del sistema es previamente conocida. Esta disparidad se define como las diferencias espaciales entre las imágenes del par estéreo y será considerada inversamente proporcional a la profundidad.

El problema de correspondencia es considerado una de las partes centrales y más difíciles del problema estéreo.

2. Desarrollo

2.1. Herramientas de desarrollo

Para el desarrollo se utilizó el lenguaje de programación Java SE; NetBeans 7.2 como entorno de desarrollo y OpenCV que es una biblioteca libre de visión artificial. Las librerías de OpenCV están programadas en C y C++; afortunadamente Google desarrolló una serie de librerías para poder implementar OpenCV en Java, y la

denominó JavaCV. Esta librería contiene métodos de visión artificial para aplicaciones en la robótica, como obtención de imágenes, procesamiento y exposición de resultados.

2.2. Hardware utilizado

Los ensayos se realizaron utilizando el siguiente hardware:

- Procesador Intel Core i5
- Placa de video con soporte para 8, 16, 24 o 32 bits.
- Cámara USB , Logitech HD Webcam C615

2.3. Montura propuesta y escenario de prueba

La montura mecánica se fabricó sobre una barra de aluminio con pares de perforaciones a lo largo de ella para poder variar las distancias entre las ópticas de las cámaras. Las cámaras están sujetas a la barra mediante un tornillo estándar WW ¼" (6,35 mm) situado por debajo de las mismas.

Se ejecutaron perforaciones 35, 50, 100 y 150, 200 y 250 mm a ambos lados del bastidor de soporte. De esta manera, la mínima distancia entre las ópticas fue de 70mm; la máxima, de 500 mm.

Se puede cambiar la altura de la montura gracias a unas barras roscadas sujetas a la barra de aluminio y a un trípode.

En el centro de la montura se colocó un nivel de burbuja para poder ajustar la alineación horizontal.



Fig. 1. Vista frontal de las cámaras sobre la montura de calibración. Las cámaras se encuentran a 150 mm del centro del sistema.

Para la alineación de las cámaras se construyó un escenario vertical de prueba con planchas de poliestireno de 4 m de ancho por 2 m de alto. Sobre esta superficie se dibujó una cuadrícula ortogonal con cuadrados de 0,2 m x 0,2 m. Este escenario se colocó verticalmente sobre la pared y de frente la montura con las cámaras web incorporadas. En el centro de la cuadrícula se trazaron referencias adicionales para calibrar los centros ópticos de las cámaras.

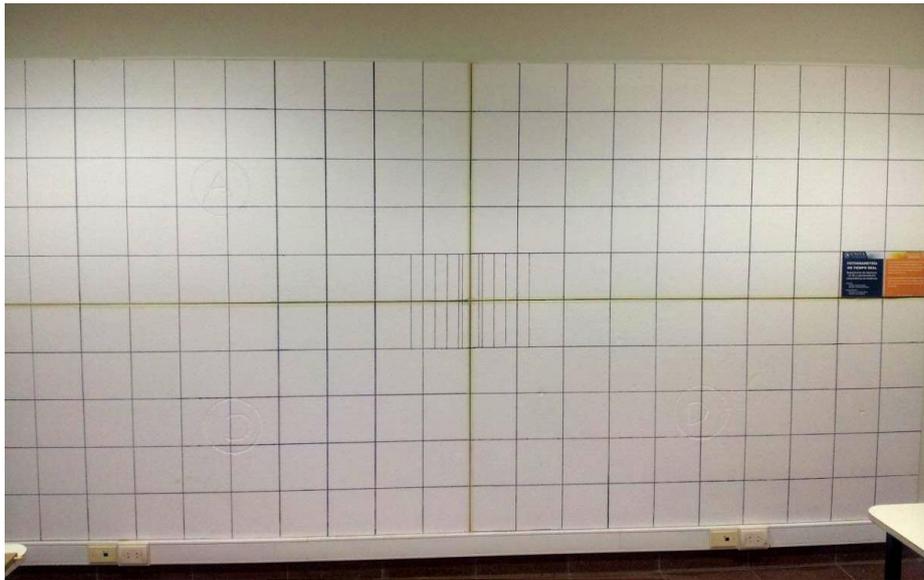


Fig. 2. Escenario de prueba y calibración.

2.4. Descripción del sistema

A continuación el esquema de la geometría del sistema de captura de imágenes estereoscópicas capturadas en el laboratorio (vista superior):

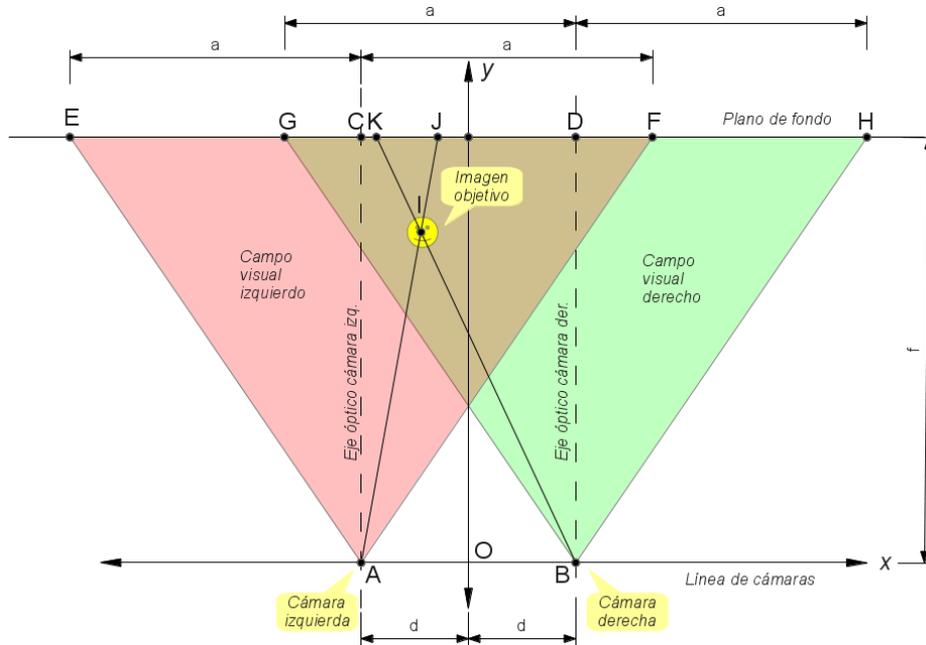


Fig. 3. Esquema geométrico del modelo.

La siguiente vista incorpora en la escena, las proyecciones de la imagen objetivo sobre el plano de fondo. El segmento AI se proyecta hasta el punto J del plano de fondo; el segmento BK , hasta el punto K . Quedan así determinadas dos rectas: AJ y BK ; su intersección, que puede determinarse como un sistema de dos ecuaciones lineales con dos incógnitas, es la ubicación de la imagen objetivo.

Los parámetros fijos del sistema son:

d , distancia entre cada cámara y el origen del sistema. Se asume que ambas están a la misma distancia. Determinado por la montura desarrollada, d puede tomar los siguientes valores fijos: 35 mm; 50 mm; 100 mm; 150 mm; 200 mm y 250 mm.

f , distancia entre la línea de cámaras y el plano de fondo. Se asume que la línea es paralela con el eje horizontal del plano. En el escenario montado para los experimentos, esta distancia se fijó en 2 825 mm.

a , apertura del campo visual, determinado por la distancia focal de cada cámara. Experimentalmente se determinó que, para la distancia f usada, a toma el valor de 1 840 mm.

r , la resolución horizontal de la cámara (en píxeles; normalmente entre 640 y 1 920)

Como restricción, la imagen (punto *I*) debe ser visible por ambas cámaras. Por lo tanto la zona útil del sistema está determinada por la intersección de los segmentos *AF* con *BG*.

Obtención de la distancia:

Habiendo establecido la geometría del sistema estereoscópico, las coordenadas del punto *I* pueden ser calculadas mediante las siguientes ecuaciones:

$$CIN = Collzq - \frac{r}{2}$$

$$CDN = ColDer - \frac{r}{2}$$

$$x_I = -d \frac{CDN + CIN}{CDN - CIN}$$

$$y_I = f r \frac{x_I + d}{2 a CIN}$$

Donde:

Collzq es el número de píxel de la columna de la imagen izquierda, donde se encuentra el objetivo, dentro del rango $0 \leq Collzq < r$

ColDer es el número de píxel de la columna de la imagen derecha, donde se encuentra el objetivo, dentro del rango $0 \leq ColDer < r$

CIN es la columna izquierda normalizada; representa la distancia desde el objetivo hasta el centro de la imagen, todavía expresada en píxeles, dentro del rango $-r/2 \leq CIN < r/2$

CDN ídem a *CIN*, para la imagen derecha

Una vez obtenidas las fórmulas para el cálculo de distancias, fue necesario desarrollar una calculadora estereoscópica para realizar varias pruebas con diferentes parámetros de entrada y determinar una configuración óptima del sistema estereoscópico para minimizar el error. La calculadora se implementó con una hoja de cálculo. Las siguientes tablas transcriben las celdas de la hoja.

Parámetros	d	150	mm	Distancia entre el centro del sistema O y cada una de las cámaras (A y B), en mm
	f	2 825	mm	Distancia entre la línea de cámaras y el fondo del escenario (AC = BD)
	a	1 840	mm	Distancia entre el centro de imagen en el fondo, y el extremo izquierdo o derecho alcanzable en la imagen (EC=CF=GD=DH)
	r	1 920	px	Resolución horizontal de la imagen, en píxeles
Datos de	FilIzq	540	px	Fila del objetivo, en la imagen izquierda, en píxeles

entrada	Collzq	1 919	px	Columna del objetivo, en la imagen izquierda, en píxeles
	FilDer	540	px	Fila del objetivo, en la imagen derecha, en píxeles
	ColDer	1	px	Columna del objetivo, en la imagen derecha, en píxeles
Resultados	Ix	0	mm	Coordenada X de la imagen, en mm
	Iy	231	mm	Coordenada Y de la imagen, en mm
	Iz		mm	Coordenada Z de la imagen, en mm
Cálculos intermedios	r2	960	px	r/2
	CJ	1 838,08	mm	Distancia CJ
	DK	-1 838,08	mm	Distancia DK
	Jx	1 688,08	mm	
	Kx	-1 688,08	mm	
	CIN	959		Col. Izq. Normalizada = Collzq - r/2
	CDN	-959		Col. der. Normalizada = ColDer - r/2

	P	X	Y	Descripción
Nomenclatura de los puntos bidimensionales	A	-150	0	Cámara izquierda
	B	150	0	Cámara derecha
	C	-150	2 825	Punto del fondo que es el centro de la imagen izquierda
	D	150	2 825	Punto del fondo que es el centro de la imagen derecha
	E	-1 990	2 825	Punto más izquierdo del fondo, visible en la imagen izquierda
	F	1 690	2 825	Punto más derecho del fondo, visible en la imagen izquierda
	G	-1 690	2 825	Punto más izquierdo del fondo, visible en la imagen derecha
	H	1 990	2 825	Punto más derecho del fondo, visible en la imagen derecha
	I	0	230,53	Posición del objeto a medir
	J	1 688,08	2 825	Punto del fondo donde se proyecta la línea AI
	K	-1 688,08	2 825	Punto del fondo donde se proyecta la línea BI
	O	0	0	Centro del sistema cartesiano, a media distancia "d" entre A y B

Si el objetivo se desplaza a lo largo del eje y , cuando se encuentre en la intersección AF con BG , se obtendrá la máxima discrepancia y la mínima distancia:

$$Collzq = r - l; ColDer = 0$$

A medida que el objetivo se aleja:

$$Collzq \rightarrow \frac{r}{2}; ColDer \rightarrow \frac{r}{2}$$

La máxima distancia observable será:

$$ColIzq = \frac{r}{2} + 1; ColDer = \frac{r}{2} - 1$$

Aprovechando la hoja de cálculo se hizo variar simétricamente ambas columnas, un píxel por vez cada una. Para cada discrepancia se calculó la distancia y se trazó el siguiente gráfico, en el que el eje de abscisas representa la variación de la discrepancia, desde cero hasta $r/2$, en píxeles; y el eje de ordenadas, la distancia del objetivo, en milímetros. Puede observarse que, al final de la excursión de la discrepancia, la distancia se dispara asintóticamente.

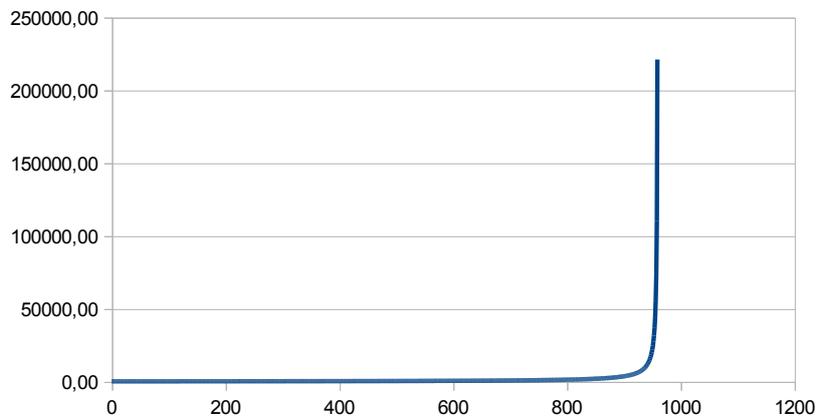


Fig. 4. Error vs. Distancia.

Como método de correspondencia se utilizó el algoritmo basados en área, en donde los elementos a comparar son ventanas de la imagen de dimensión fija, y el criterio de semejanza es una medida de la correspondencia entre las ventanas de las dos imágenes. El elemento correspondiente queda determinado por la ventana que maximiza el criterio de semejanza dentro de la región de búsqueda.

El coeficiente de correlación "C" está definido por la siguiente fórmula:

$$C = \sum_{i=1}^M \sum_{j=1}^N \left(\frac{L_I(i,j)}{\sqrt{\frac{\sum_{i=0}^2 (P_i - \bar{P}_i)^2}{2}}} - \frac{L_D(i,j)}{\sqrt{\frac{\sum_{i=0}^2 (P_i - \bar{P}_i)^2}{2}}} \right)^2$$

Con esta fórmula se desarrolló el método

buscarPunto(Image imgIzquierda, Image imgDerecha, int x ,
int y, int ventana)

Éste método se encarga de asociar a un punto dado de la imagen izquierda a su correspondiente en la imagen derecha.

Como el método es necesario que funcione en tiempo real se implementó para la correspondencia de puntos dos métodos más que ayudan a optimizar el algoritmo. Uno es de seguimiento de objetos (*tracking*) y el otro es de auto actualización de objetos, que ayudará al *tracking* cuando el objeto que se está buscando cambie de perspectiva.

Teniendo los parámetros de las cámaras y el mapa de disparidad con la calculadora estereoscópica desarrollada es directa la generación del mapa de profundidad.

2.5. GUI (Interfaz gráfica del usuario)

Pantalla Principal



Fig. 5. Pantalla Principal

Consta de tres botones:

- Configuración: Permite configurar los parámetros del sistema estereoscópico.
- Adquisición: Corresponde a la tarea de correspondencia de puntos en el par estereoscópico. Mediante la selección de un objeto calcula la distancia que hay hacia el lente de la cámara.
- Mapa: Se muestra el mapa de profundidad (DEM) representado mediante colores.

Pantalla de configuración

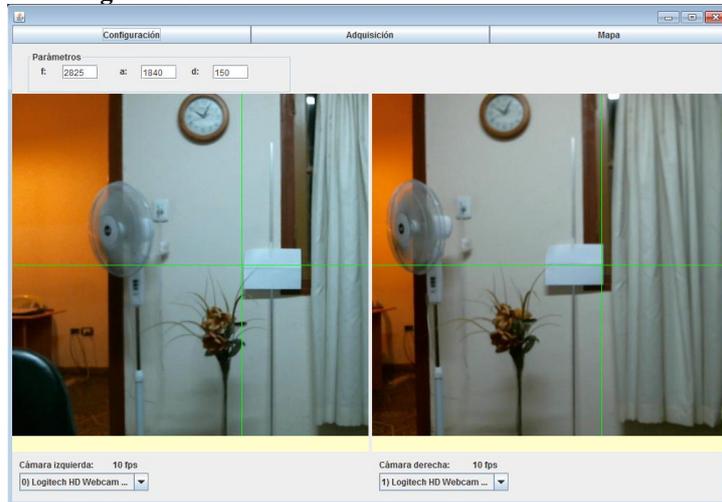


Fig. 6. Pantalla de configuración

Aquí se puede ingresar los parámetros del sistema estereoscópico como f (distancia entre la línea de cámaras y el plano de fondo), a (apertura del campo visual), d (distancia entre cada cámara y el origen del sistema), como también se puede seleccionar las cámaras mostrando las vistas correspondientes con sus respectivos cuadros por segundo (FPS). También se le incorporó ejes en las 2 proyecciones de la cámara para ayudar a su calibración.

Pantalla de Adquisición

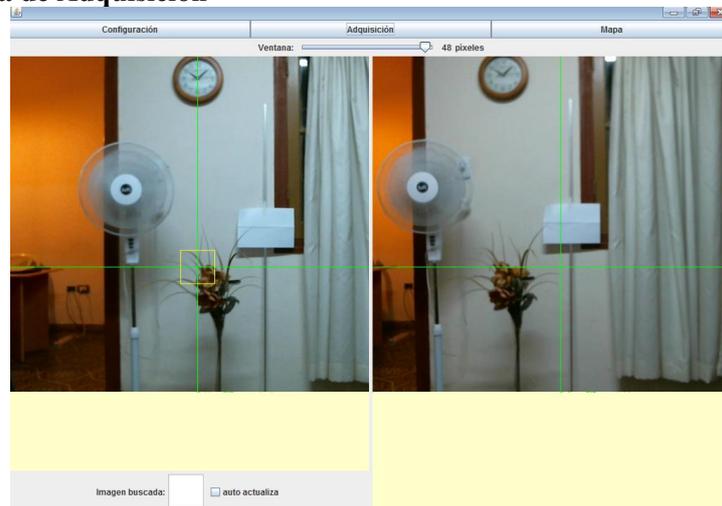


Fig. 7. Pantalla de Adquisición

En esta pantalla se implementa la correspondencia de puntos con seguimiento de objetos (*tracking*).

En la parte superior se puede regular el tamaño de la ventana a seleccionar, que posteriormente sobre la imagen izquierda podremos seleccionar el objeto a seguir.

Una vez seleccionado el objeto será marcado con una ventana de color celeste mostrando su distancia aproximada con respecto a las cámaras expresada en milímetros y rastreado en las dos imágenes como muestra la fig. 7. En la parte inferior se muestra una vista previa del objeto seleccionado. Por último para optimizar el seguimiento de un objeto, se agregó una opción de auto actualización, ya que un objeto muchas veces en su movimiento puede cambiar de perspectiva o rotación, con esta opción mejora ampliamente la búsqueda.

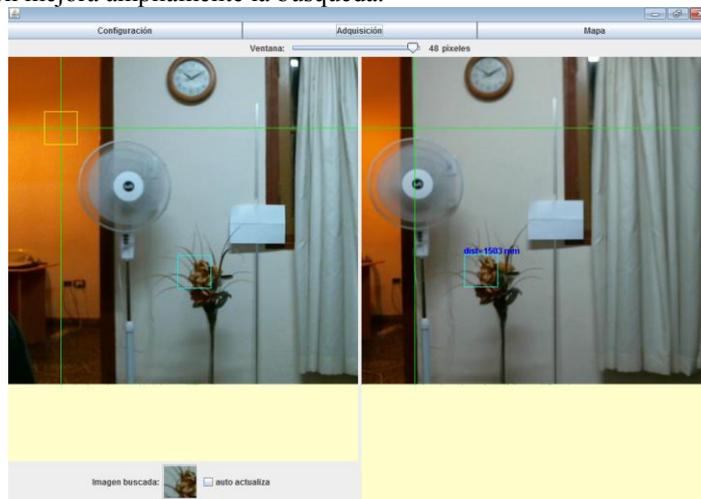


Fig. 8. Pantalla de Adquisición con tracking de objetos

Pantalla de Mapa

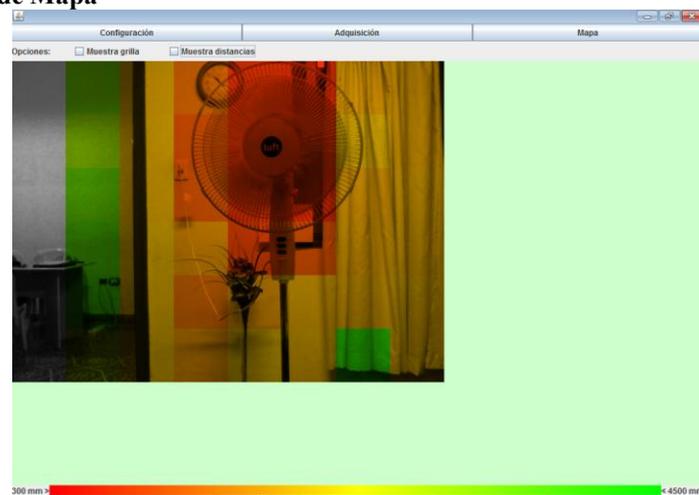


Fig. 9. Pantalla de mapa de disparidad

Usando repetidas veces el método de la calculadora estereoscópica, se puede identificar cada ventana de la pantalla con un color de a proximidad de la escala cromática en la parte inferior. Representando con el color rojo una proximidad de 300 mm y con el verde una proximidad de 4 500 mm.

Para obtener una mejor apreciación de los valores capturados de cada ventana se puede tildar la opción “Muestra distancia”, esto mostrará en todas las ventanas una distancia aproximada en milímetros que está tomando el método de correspondencia en los distintos sectores como se muestra en la fig. 2.6.

Cuenta también esta pantalla con una opción de visualización de grilla que separa todos los sectores de la pantalla teniendo una mejor vista de las ventanas que conforman el mapa de profundidad.



Fig. 10. Pantalla de Mapa de disparidad, con visualización de distancias y grilla

3. Conclusión

Los objetivos del proyecto fueron cumplidos con éxito bajo las condiciones impuestas de hardware, lenguaje de desarrollo e implementación en tiempo real.

Para la “Telemetría en pares estereoscópicos” se logró obtener un algoritmo óptimo de *tracking* en escenarios controlados, con buena iluminación y objetos con contrastes bien definidos. Pero como contraposición a un escenario controlado, cuando carece de suficiente luminancia y contraste el algoritmo se confunde de patrones marcando objetos erróneos y por lo tanto arrastrando el error al método de correspondencia y posteriormente a la calculadora estereoscópica.

La generación del mapa de profundidad (DEM) es computacionalmente lento por falta de hardware.

La secuencia que realiza el programa para la generación del DEM son los siguientes:

Identificación de todas las ventanas de la imagen

- 1- Aplicar método de correlación de ventanas
- 2- Aplicar método de cálculo de distancia
- 3- Teñir cada ventana con la escala cromática de a proximidad.

Analizar las dos imágenes completas de cada cámara, requiere mucho procesamiento de datos para una representación en tiempo real.

Otro punto importante a tener en cuenta es que se implementaron métodos de calibración en tiempo real, por lo que la calibración manual realizada es susceptible a errores por movimiento de la montura estereoscópica. Este punto podría ser uno objetivos futuros a implementar para mejorar las capturas de las imágenes.

4. Referencias

- ST Barnard y Fischler MA, Stereoscopia Computacional. ACM Computing Surveys, vol.14, pp 553-572, 1982.
- O. Veksler., programación dinámica en árbol de Correspondencia estéreo. CVPR2005.
- “Segmentación por Umbralizacion.” Argentina, Cátedra de visión artificial Octubre 2005.
- C. Platero “Introducción a la Visión Artificial – Apuntes de visión artificial”, Universidad Politécnica de Madrid, España, 2009.
- “Cátedra de Visión Artificial – Iluminación para las aplicaciones de Visión Artificial” Universidad Nacional de Quilmes, Argentina, 2005.
- Librería OpenCV (Open Source Computer Vision)
<http://opencv.willowgarage.com/wiki/>.
- K. Takita, T. Aoki, M.A. Muquit y T. Higuchi. A Sub-pixel Correspondencia técnica de búsqueda para aplicaciones de visión por computador. IEICE Trans. Fundamentos, vol.E87-A, 59Referencias n° 8, agosto de 2004.
- J.Lotti y G. Giraudon. Correlación con la ventana Algoritmo Adaptativo para la imagen aérea en la visión estéreo. Inst. Nac.. de Investigación en inf. et Autom., Shophia-Antipolis. Conferencia Internacional sobre Reconocimiento de Patrones, 1994.
- M. Pollefeys, R. Koch y L. Van Gool. A simple and efficient rectification method for general motion. IEEE International Conference on Computer Vision, (1): 496-501, 1999.
- Emanuele Trucco y Alessandro Verri, Introductory Techniques for 3D Computer Vision. Prentice Hall,1998.
- R.F. Lecumberry. Cálculo de disparidad y segmentación de Objetos en secuencias de video. Tesis de Maestría en Ingeniería Eléctrica. Montevideo, Uruguay, Agosto 2005.
- Daniel Scharstein y Richard Szeliski. Taxonomía y evaluación de las densidad de dos tramas estéreo Algoritmos correspondencia. Revista Internacional de Visión por Computador, pp 7-42, abril-junio de 2002.
- Deitel Harvey M, Deitel Paul J.- Cómo Programar en Java -Deitel 7ma edición - año 2008.