

LOCALIZACIÓN ESPACIAL DE UN PUNTO EN XYZ MEDIANTE VISIÓN ARTIFICIAL

SPACE LOCATION OF A POINT USING ARTIFICIAL VISION

Fecha de Recepción: 16 de mayo de 2006

Fecha de Aprobación: 24 de julio de 2006

Ricardo Andrés Castillo¹

Iván Andrés Díaz López²

Mauricio Huertas León³

RESUMEN

Con el fin de localizar un punto en el espacio mediante el uso de cámaras Web (visión artificial), se tuvieron en cuenta los siguientes aspectos: parámetros de la cámara, tipo de proceso a efectuar sobre la imagen capturada y presentación de los resultados. Con el objetivo de encontrar el campo útil de visión de la cámara, se determinó experimentalmente el ángulo para el cual la distorsión radial es mínima; luego por medio de una red neuronal tipo Perceptrón Multicapa de 3 entradas, 7 neuronas ocultas y 10 salidas, se filtra la imagen procedente de cada una de las cámaras identificando en estas un color específico; promediando los puntos de dispersión resultantes, se ubica bidimensionalmente en cada imagen el centroide del objeto, con base en esta información y aplicando un desarrollo matemático, se localiza completamente el objeto en un sistema de coordenadas aleatorio XYZ, respecto del cual, inicialmente, solo se necesita conocer la posición de cada una de las cámaras. Se desarrolló un software para computador, que permite observar las imágenes de cada una de las cámaras y la configuración de la mayoría de parámetros. En la calibración del sistema se deben tener en cuenta: orientación de las cámaras, distancias sugeridas entre los focos de estas y calidad de las imágenes obtenidas.

Palabras clave: Filtro, procesamiento de imágenes, red neuronal, visión artificial.

ABSTRACT

With the purpose of locating a point in the space using WebCams (artificial vision), the following aspects were considered: camera parameters, type of process to carry out on the captured image and presentation of the results. With the objective to find the camera useful field of vision, the angle for which the radial distortion is minimum was experimentally determined; then using a Multilayer Perceptron neural network with 3 inputs, 7 hidden neurons and 10 outputs, the images from each camera are filtered to identify a specific color; averaging the resulting set of points, the center of the object is bidimensionally located in each image, with this information and applying a mathematical development, the center of the object is completely located in a random XYZ orthogonal axis, respect to which, is only needed to know the position of each camera. A computer software, that allows to observe the images of each camera and the configuration of most of parameters, was developed. In the system calibration is necessary to consider: cameras direction, suggested distances between the centers of cameras and quality of the obtained images.

Key Words: Filter, Image Processing, Neural Network, Artificial Vision.

¹ Ingeniero en Mecatrónica, Universidad Militar "Nueva Granada". Docente - Investigador grupo DAVINCL E – mail: ricardo.castillo@umng.edu.co.

² Ingeniero en Mecatrónica, Universidad Militar "Nueva Granada".

³ Ingeniero en Mecatrónica, Universidad Militar "Nueva Granada".

I. INTRODUCCIÓN

Por medio de la visión artificial se han desarrollado estudios y adelantos que permiten el avance de la ciencia y la tecnología, sin la necesidad de elementos u operadores presentes que puedan interferir en el medio de trabajo.

Hoy en día gracias a la aplicación de tecnologías en informática y computación como las redes neuronales, la visión artificial y la lógica difusa, se han llevado a cabo la actualización de diferentes tipos de procesos tanto industriales como administrativos, los cuales traen como beneficio la reducción del tiempo en los procesos. Estos nuevos avances han facilitado el trabajo del hombre haciéndolo más eficiente.

El propósito de este trabajo es desarrollar una herramienta basada en visión artificial que puede ser aplicada en áreas como: robótica de manipuladores, navegación de robots móviles y teleoperación entre otras.

Los sistemas de localización SAP⁴, son construidos en su mayoría con elementos mecánicos, por medio de los cuales la adquisición de información se debe únicamente a la fidelidad de un sistema de palpado, aunque son sistemas eficaces su operación es muy lenta debido a que se requiere coordinar los motores, de manera tal que el sistema se acerque a su objetivo de la forma más apropiada; para evitar esta serie de inconvenientes, se utilizará visión artificial, en este caso la medición es mucho más rápida y las estructuras necesarias son más sencillas. Evitando algunas partes mecánicas como brazos y motores, el sistema es menos costoso y su tamaño es reducido, por lo cual se podría usar en lugares estrechos. La Figura 1 muestra un sistema SAP sin contacto.



Figura 1. Maquina UMM500 [1]

II. MÉTODOS

A. Teoría de la Visión de la Cámara y la Forma de la Composición de las Imágenes Partir de los Ángulos.

La visión de una cámara simula la visión de un ojo humano. Las restricciones en las cámaras están dadas por la lente que poseen estos dispositivos, definiéndose en estas un ángulo de visión.

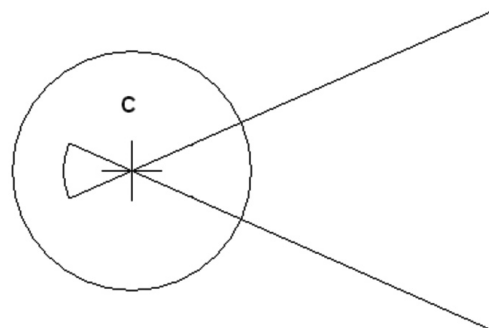


Figura 2. Vista superior del sistema, plano XY

⁴ Sistema de Adquisición de Puntos

La Figura 2 muestra el máximo ángulo de visión de una cámara. Se realizaron varias pruebas para determinar estos ángulos en cada cámara y se obtuvo el mismo ángulo a una distancia de 78 cm. Los ángulos fueron obtenidos mediante la siguiente ecuación:

$$\alpha = 2 * \tan^{-1} \left(\frac{L}{2 * H} \right) \quad (1)$$

Donde L es la distancia barrida por la cámara y H es la distancia entre la cámara y la superficie como se muestra en la Figura 3.

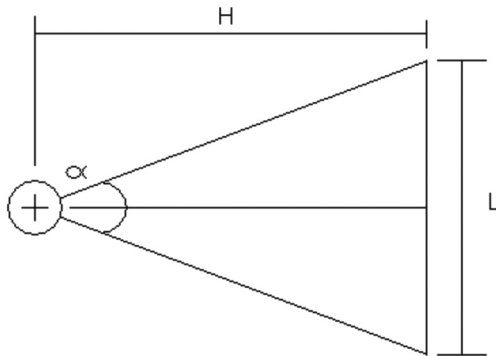


Figura 3. Disposición de la cámara para determinar el ángulo de visión

Con esta información se pueden determinar los ángulos para cada resolución de la cámara (160x120 y 320x240); estos datos son luego ingresados al software que determina la posición final del sistema.

Al realizar una simulación con las líneas proyectadas por el objeto y recibidas por la cámara, se puede hacer una aproximación de la formación de una imagen con dichas líneas. La Figura 4 ilustra este concepto.

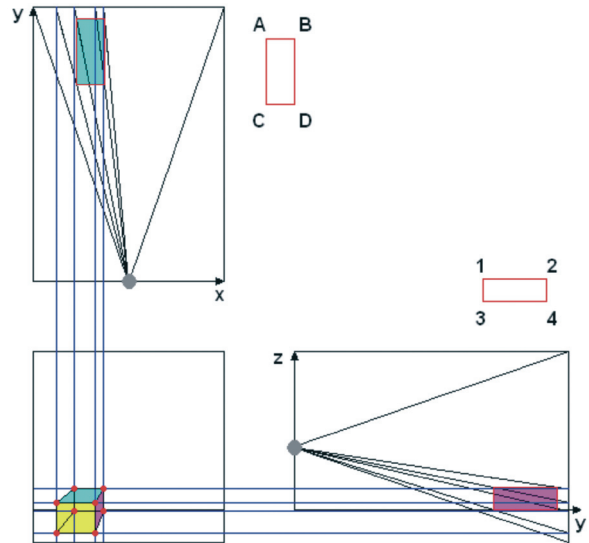


Figura 4. Proyección de ángulos para lograr una imagen con perspectiva

Como se puede ver, los puntos de la imagen superior izquierda en la Figura 4, se relacionan con los puntos de la imagen inferior derecha para formar la perspectiva de la siguiente manera:

Tabla 1. Unión de puntos para formar la perspectiva

A	2
A	4
B	2
B	4
C	1
C	3
D	1
D	3

Las líneas de proyección de los ángulos se representan en la imagen del computador por la posición X y Y de cada píxel permitiendo determinar los ángulos de visión, θ_1 y θ_2 , como se observa en la Figura 5.

$$\theta_1 = \tan^{-1} \left(\frac{(2X - \text{Ancho Imagen}) \tan \left(\frac{\text{AnguloMaximoX}}{2} \right)}{\text{Ancho Imagen}} \right) \quad (2)$$

$$\theta_2 = \tan^{-1} \left(\frac{(\text{Alto Imagen} - 2Y) \tan \left(\frac{\text{AnguloMaximoY}}{2} \right)}{\text{Alto Imagen}} \right) \quad (3)$$

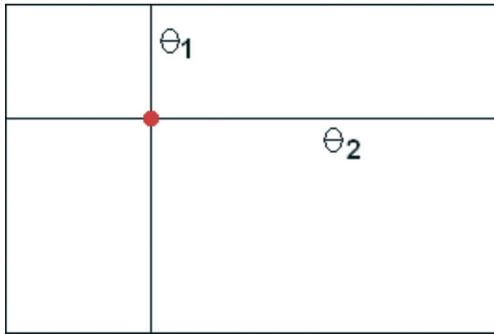


Figura 5. Coordenadas Angulares del Punto

En las ecuaciones 2 y 3, el ancho y el alto de la imagen están dados para cada una de las resoluciones, además los ángulos determinados en cualquier caso, tanto de arriba hacia abajo, como de un lado al otro se conocen. Los valores de X, Y, alto y ancho están en píxeles y los ángulos en grados.

B. Filtración de las Imágenes y Determinación del Punto a Evaluar

El punto a evaluar se encuentra dentro de un espacio tridimensional, en el cual pueden estar presentes varios elementos además del objetivo; se debe realizar un filtrado sobre la imagen para eliminar aquellos objetos que no sean de ningún interés.

Cada punto de la imagen esta compuesto por tres canales o tonos que combinados

forman cualquier color; existen diferentes sistemas de colores entre los cuales encontramos CMY (Cian, Magenta, Yellow), CMYK (Cian, Magenta, Yellow, Key), HSV (Hue, Saturation, Value), HSI (Hue, Saturation, Intensity), RGB (Red, Green, Blue), entre otros [8], ver Figura 6. Cada uno de estos sistemas se relaciona con los demás por medio de transformaciones matemáticas.

Para definir el tipo de filtro que se implementaría a la imagen se plantearon diferentes soluciones, inicialmente se implementaron filtros con máscaras, no obstante, durante las pruebas, al variar el ambiente, las luces y las sombras, no se obtuvieron resultados satisfactorios. Posteriormente se implementó un filtro definiendo ciertos colores deseados para encontrarlos en la imagen, al definir rangos en cada uno de los componentes R,G,B; en este caso la información importante en la imagen se perdía frecuentemente. Las soluciones planteadas anteriormente, dan el mismo resultado si usamos los sistemas RGB o CMY.

Se desarrollo otro filtro usando la transformación del sistema RGB al sistema HSV, el cual no tiene la dificultad de definir rangos en 3 canales para un color específico, en vez de esto, es suficiente con determinar el rango en un solo canal, la tonalidad: es un valor en grados y representa un ángulo de rotación en este sistema, de esta forma, se trabaja definiendo un rango en este ángulo; por ejemplo, el rojo se puede tomar como un rango entre -30° y 30° .

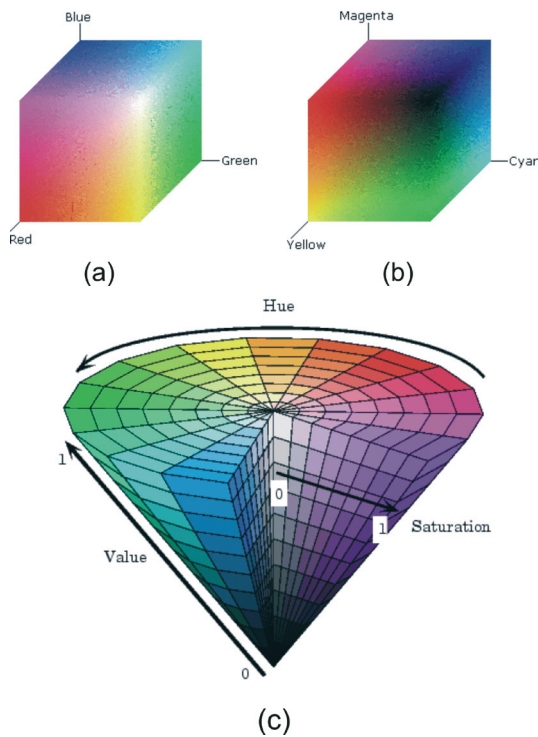


Figura 6. Espacio de colores en tres sistemas (a) RGB, (b) CMY y (c) HSV [6]

El filtro implementado con el sistema de colores HSV, aunque mejoró los resultados de los otros filtros, presenta problemas al cambiar las condiciones de iluminación; debido a esto, se decidió realizar una red neuronal.

En el filtro implementado con la red neuronal, el número de entradas es de tres (3), en la capa oculta se tienen siete (7) elementos y en la salida se pueden identificar diez (10) colores.

En la red neuronal implementada, las entradas corresponden a los canales del sistema RGB, cuyos valores pasan por una función que escala que los lleva desde [0,255], a valores normalizados en el intervalo [-1,1]. Como podemos ver en la Figura 8; además de las entradas RGB existe una entrada de activación que usualmente toma el va-

lor de +1 y elimina posibles inconvenientes en la salida cuando las entradas son cero [2].

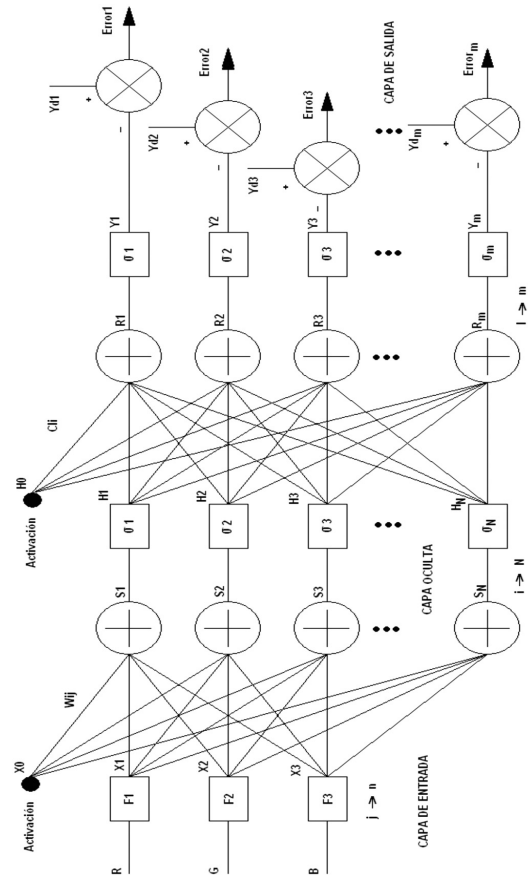


Figura 7. Esquema general de la red neuronal implementada [5]

Para ajustar los parámetros de la red MLP, de tal forma que esta responda de la manera deseada identificando ciertos colores definidos, se debe realizar un entrenamiento, para ello se utilizó el método de propagación inversa (Back Propagation). Este método de entrenamiento permite ajustar los pesos de la red neuronal, luego de presentar repetidamente un conjunto de patrones entrada-salida deseada [3].

Ya que las salidas del MLP están dentro de un rango de $[-1,+1]$, se debe efectuar un procedimiento de escala sobre estos datos para visualizar la salida total en porcentaje [5], indicando la similitud de cada uno de los colores de entrada con los colores deseados definidos con anterioridad. Para decidir si algún color de entrada corresponde a alguno de los colores deseados se especifica un umbral para cada salida, este umbral es un parámetro modificable en el software en un rango de $[80,90]$.

Las salidas del MLP en su orden 1.rojo, 2.verde, 3.azul, 4.amarillo, 5.magenta, 6.cyan, 7.negro, 8.blanco, de estas, las dos últimas se encuentran a disposición para otro color o tono que se desee entrenar. Hay que tener en cuenta que los tonos propuestos no son fijos, es decir se pueden asignar a conveniencia de quien implemente el sistema. El filtrado arroja una imagen con una dispersión de puntos, que en caso óptimo estarán agrupados en una sola sección de la imagen. Para encontrar un solo punto a evaluar dentro de este conjunto, se realiza un promedio de las distancias desde una referencia (en este caso el origen de coordenadas de la imagen o esquina superior izquierda), de cada una de las partículas o puntos encontrados después de aplicado el filtro.

C. Definición de las Coordenadas XYZ desde la Perspectiva de la Cámara

El ángulo θ_1 , en la Figura 8, es el ángulo comprendido entre la recta perpendicular al eje X que pasa por el centro de la cámara 1 (C1) donde tendrá un valor de 0° y la recta

que forma el centro de la cámara 1 (C1) y el punto (P). El ángulo tiene valores positivos en sentido de las manecillas del reloj y valores negativos en sentido contrario a las manecillas del reloj.

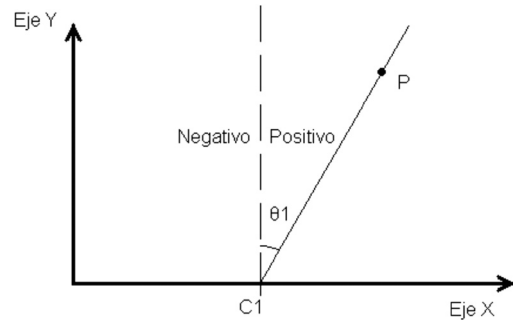


Figura 8. Vista superior del sistema, plano XY cámara 1

El ángulo θ_2 , en la Figura 9, es el ángulo comprendido entre la recta perpendicular al eje X que pasa por el centro de la cámara 2 (C2) donde tendrá un valor de 0° y la recta que forma el centro de la cámara 2 (C2) y el punto (P). El ángulo tendrá valores positivos en sentido de las manecillas del reloj y valor negativo en sentido contrario de las manecillas del reloj.

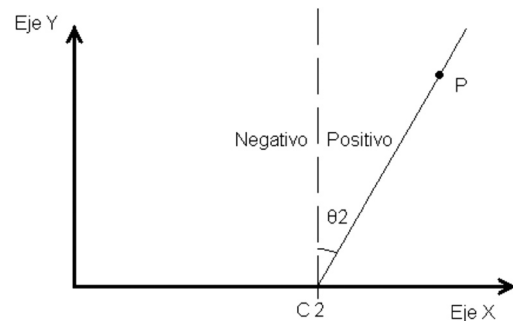


Figura 9. Vista superior del sistema, plano XY cámara 2

Los centros de las cámaras están separados por una distancia d_1 medida sobre el eje X, ver Figura 10, la cual es determinada por las dimensiones de las cámaras.

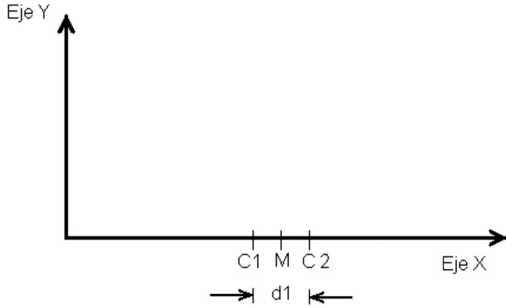


Figura 10. Vista superior del sistema, plano XY esquemático

El ángulo θ_3 , en la Figura 11, es el ángulo comprendido entre la recta perpendicular al eje Z que pasa por el centro de las cámaras (C1 y C2) donde tendrá un valor de 0° y la recta que forma el centro de la cámara 1 (C1) y el punto (P). El ángulo tendrá valores positivos en sentido contrario de las manecillas del reloj y valor negativo en sentido de las manecillas del reloj.

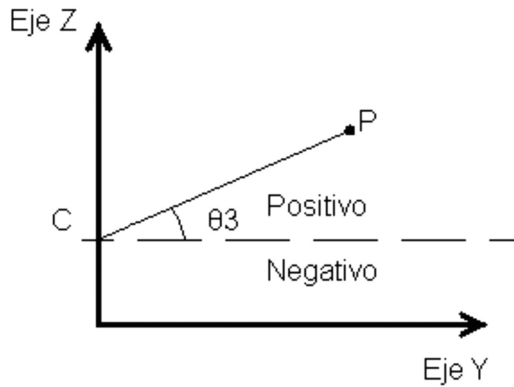


Figura 11. Vista lateral del sistema, plano YZ

Realizando un proceso de triangulación, teniendo como entrada los ángulos θ_1 , θ_2 y θ_3 , y utilizando la ley del seno y del coseno, encontramos las coordenadas P_x , P_y y P_z del punto P, como se describe a continuación en la ecuación 4. Las distancias d_1 y M son medidas a lo largo del eje X.

$$\begin{aligned}
 P_x &= M + \frac{d_1}{2} + \frac{d_1 \cos(\theta_1) \sin(\theta_2)}{\sin(\theta_1 - \theta_2)} \\
 P_y &= \frac{d_1 \cos(\theta_1) \cos(\theta_2)}{\sin(\theta_1 - \theta_2)} \\
 P_z &= d_2 + \frac{d_1 \cos(\theta_1) \cos(\theta_2) \tan \theta_3}{\sin(\theta_1 - \theta_2)}
 \end{aligned} \tag{4}$$

Al replantear las ecuaciones para ubicar el origen del sistema coordenado en el centro de las cámaras obtenemos:

$$\begin{aligned}
 P_x &= \frac{d_1}{2} + \frac{d_1 \cos(\theta_1) \sin(\theta_2)}{\sin(\theta_1 - \theta_2)} \\
 P_y &= \frac{d_1 \cos(\theta_1) \cos(\theta_2)}{\sin(\theta_1 - \theta_2)} \\
 P_z &= \frac{d_1 \cos(\theta_1) \cos(\theta_2) \tan \theta_3}{\sin(\theta_1 - \theta_2)}
 \end{aligned} \tag{5}$$

Al realizar este cambio se debe tener cuidado en el signo de los ángulos, no tomando valores absolutos. El proceso de triangulación entre las 2 cámaras en el plano XY se puede apreciar en la Figura 12.

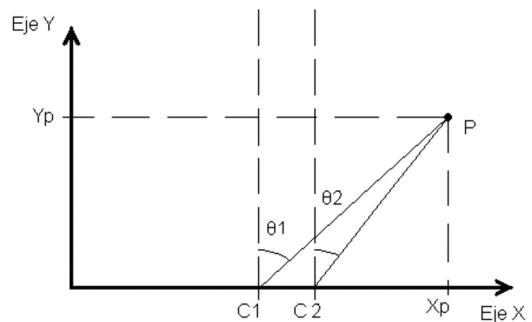


Figura 12. Triangulación del punto desde las cámaras

D. Posicionamiento de la Cámara y Generación de Coordenadas a Partir de su Posición Relativa.

Con apoyo de la robótica de manipuladores y utilizando matrices de rotación y traslación, se relaciona el sistemas de referencia ubicado en medio de las cámaras, con el sistema de referencia ubicado en un punto aleatorio; las coordenadas del primer sistema mencionado respecto al segundo son (X_o, Y_o, Z_o) [7], así se obtiene la primera matriz:

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & X_o \\ 0 & 1 & 0 & Y_o \\ 0 & 0 & 1 & Z_o \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Luego se realizan las rotaciones respecto a las nuevas coordenadas teniendo en cuenta lo siguiente:

- El ángulo α es la rotación respecto al eje Z con orientación positiva desde el eje X hacia el eje Y.
- El ángulo β es la rotación respecto al eje Y con orientación positiva desde el eje Z hacia el eje X.
- El ángulo λ es la rotación respecto al eje X con orientación positiva desde el eje Y hacia el eje Z.

Con estas rotaciones la segunda matriz de transformación es:

$$M_2 = \begin{bmatrix} C\alpha C\beta & C\alpha S\beta S\gamma - S\alpha C\gamma & C\alpha S\beta C\gamma + S\alpha S\gamma \\ S\alpha C\beta & S\alpha S\beta S\gamma + C\alpha C\gamma & S\alpha S\beta C\gamma - C\alpha S\gamma \\ -S\beta & C\beta S\gamma & C\beta C\gamma \\ 0 & 0 & 0 \end{bmatrix} \quad (7)$$

Desde este punto, se dejan ubicadas las cámaras, con posibilidad de tomar cualquier configuración que el usuario necesite para facilitar la visión del punto o para obtener perspectivas diferentes de las cámaras. Al multiplicar las matrices M1 y M2 se tiene la transformación total entre el

sistema aleatorio y el sistema coordinado de las cámaras.

Finalmente, teniendo en cuenta las coordenadas del punto P, medidas desde el sistema de referencia de las cámaras (P_x, P_y, P_z) , se obtiene:

$$M_3 = \begin{bmatrix} 1 & 0 & 0 & P_x \\ 0 & 1 & 0 & P_y \\ 0 & 0 & 1 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

Multiplicando estas 3 matrices podemos calcular las coordenadas del punto P (X, Y, Z) respecto a un sistema de coordenadas cartesiano aleatorio. De esta manera, las coordenadas del punto con respecto a la referencia estarán dadas por la cuarta columna de la matriz de transformación total. El vector que describe la posición del punto P es:

$$P_{FINAL} = \begin{bmatrix} X_p C\alpha C\beta + Y_p (C\alpha S\beta S\gamma - S\alpha C\gamma) \\ + Z_p (C\alpha S\beta C\gamma + S\alpha S\gamma) \\ + \frac{d_1}{2} + \frac{d_1 \cos(\theta_1) \text{Sen}(\theta_2)}{\text{Sen}(\theta_1 - \theta_2)} \\ \\ X_p S\alpha C\beta \\ + Y_p (S\alpha S\beta S\gamma + C\alpha C\gamma) \\ + Z_p (S\alpha S\beta C\gamma + C\alpha S\gamma) \\ + \frac{d_1 \cos(\theta_1) \text{Cos}(\theta_2)}{\text{Sen}(\theta_1 - \theta_2)} \\ \\ - X_p S\beta \\ + Y_p C\beta S\gamma \\ + Z_p C\beta C\gamma \\ + \frac{d_1 \cos(\theta_1) \text{Cos}(\theta_2) \tan \theta_3}{\text{Sen}(\theta_1 - \theta_2)} \\ 1 \end{bmatrix} \quad (9)$$

E. Implementación de la Teoría en un Sistema Real.

Para comprobar la teoría desarrollada anteriormente, se implementó un software de captura de imágenes, que utiliza la información proveniente de dos cámaras conectadas (capturando simultáneamente), para localizar un punto dentro un espacio de trabajo tridimensional. Además se realizó un montaje físico con las cámaras ubicadas en soporte tipo trípode, ver Figura 13.

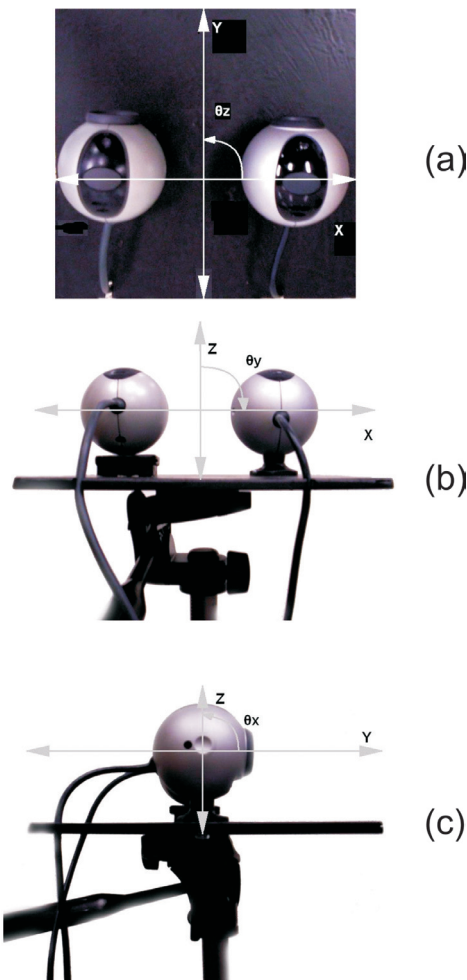


Figura 13. Representación de los planos (a) XY, (b) XZ y (c.) YZ con la descripción de las rotaciones

Para llevar a cabo la captura adecuada de las imágenes, se realizó un montaje con un soporte tipo trípode sobre el que se ubicó una placa, el cual permite los movimientos de rotación de la placa sobre los tres ejes, el desplazamiento sobre el eje Z esta dado por la función del trípode. Los desplazamientos sobre los ejes X y Y son variados con la posición de trípode sobre el suelo.

Con respecto a la cara superior de la placa, el plano XY está elevado a una distancia determinada por la construcción de las bases de las cámaras y la altura de la cámara hasta su centro. Se observa en la Figura 14, la forma adecuada de tomar la medición desde el punto de referencia.

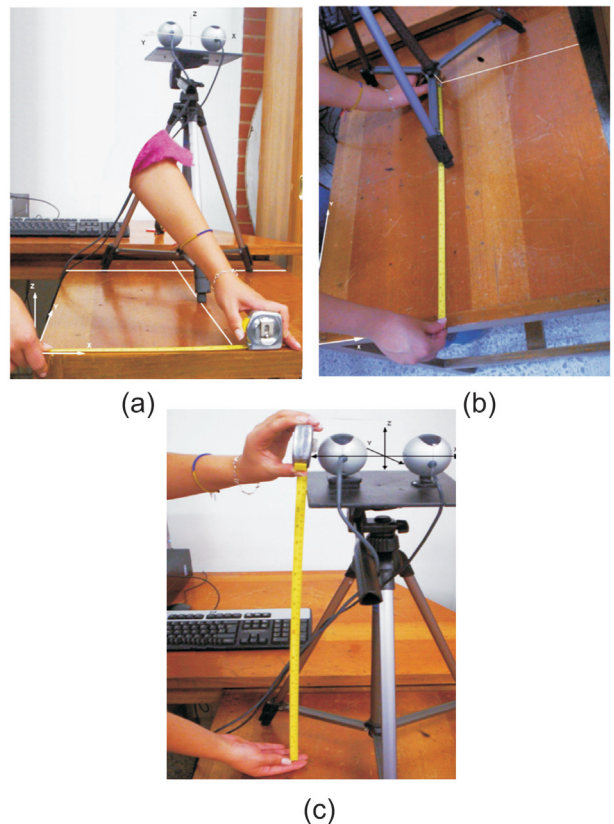


Figura 14. Forma adecuada para medir la translación (a) X, (b) Y y (c.) Z

Una vez posicionado y orientado el sistema, se procede a calibrar las cámaras buscando que las líneas de visión estén paralelas entre sí y el eje Y, sobre el plano XY. Con este propósito, se diseñó una imagen para corregir la posición de la cámara sobre su propio soporte.

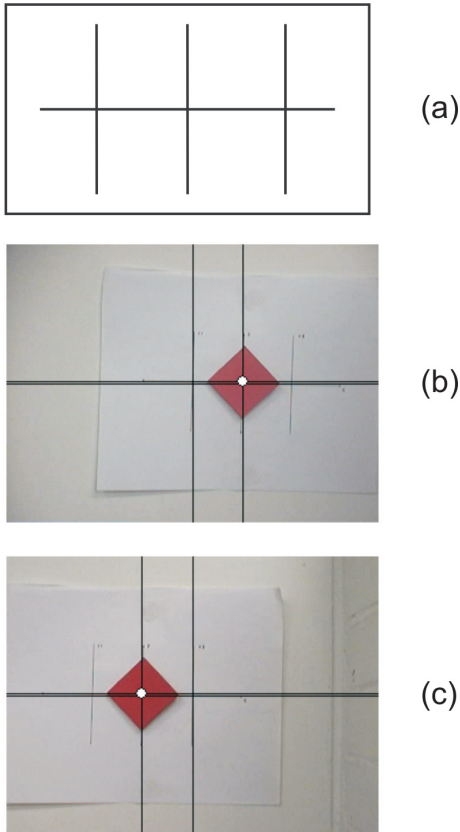


Figura 15. (a) Imagen para calibrar la posición de las cámaras, (b) vista desde la primera cámara y (c.) vista desde la segunda cámara

La calibración de los ejes por medio de la imagen se logra al alinear la gráfica como se muestra en la Figura 15. Una vez posicionado todo el sistema, se empiezan a capturar imágenes para realizar el procesamiento sobre ellas, se ubican los objetos en diferentes posiciones dentro del espacio de visión útil de ambas cámaras. Tanto el software

de captura como el software de entrenamiento de la red neuronal, abren por sí solos un archivo de nombre Red.red, el cual posee la información de los pesos de la red, el número de patrones guardados y los valores de estos con sus salidas deseadas. Al cerrar el programa de entrenamiento este guarda los cambios realizados en la red. El diseño de este software fue basado en la teoría desarrollada matemáticamente y se implementó en el lenguaje C++ de programación usando el software Builder C++ versión 6.0, además de un componente especial para el trabajo de captura, procesamiento y tratamiento de imágenes y video : VideoLab versión 2.2.

El software de las cámaras se divide en 6 partes: ConFiguración general, visualización de la cámara 1, visualización de la cámara 2, muestra de resultados gráficos, muestra de resultados numéricos y ayuda del programa. Cada una de estas partes se puede manejar de una manera cómoda y fácil, permitiendo a cualquier persona que conozca el propósito del software un acceso bien logrado dentro del mismo. Ver Figura 16.

Existe un menú que se encuentra activo sin importar la pestaña seleccionada, en el cual esta el control para empezar la captura, detenerla, pasar el fotograma visualizado y buscar en esa imagen el punto deseado, así como las opciones de dejar la imagen de fondo al buscar en la imagen, poner una referencia en la imagen y la selección del color que se desea encontrar. Al realizar estas operaciones, se trabaja simultáneamente sobre las dos cámaras; al buscar se muestran las coordenadas en píxeles del centro de masa de la dispersión encontrada y el valor del ángulo correspondiente.

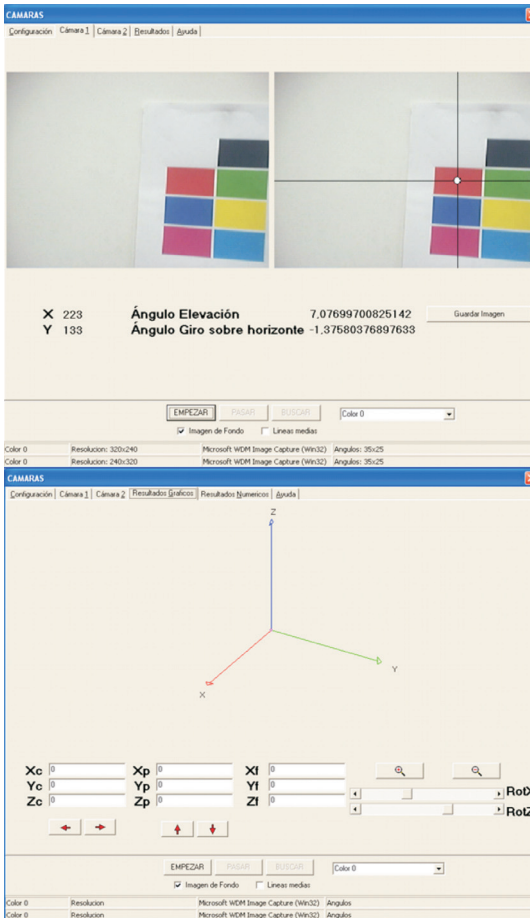


Figura 16. Algunas de las pestañas del programa (a) Muestra de imágenes de la cámara y (b) Muestra de resultados gráficos

III. PRUEBAS DE CAMPO

A continuación se muestra el proceso a seguir para realizar las pruebas de campo.

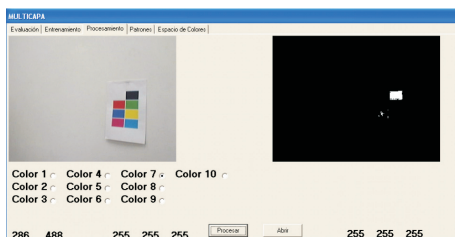


Figura 17. Imagen antes y después de filtro

En la Figura 17 se observa a la izquierda la imagen tomada de la cámara 1 y a la derecha el resultado del filtro para el color 7 (negro), en la imagen se observa ruido en la parte del color 3 (azul). Utilizando el programa de entrenamiento de la red neuronal, se corrigen estos errores ingresando nuevos patrones o muestras. Figura 18.

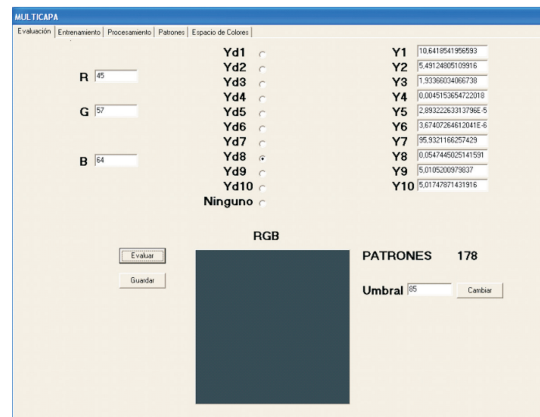


Figura 18. Ventana de Evaluación

El proceso de entrenamiento de la red neuronal puede ser automático o por pasos según la elección y terminará automáticamente cuando el error alcance un valor por debajo al especificado (0,02 preferiblemente). Figura 19.

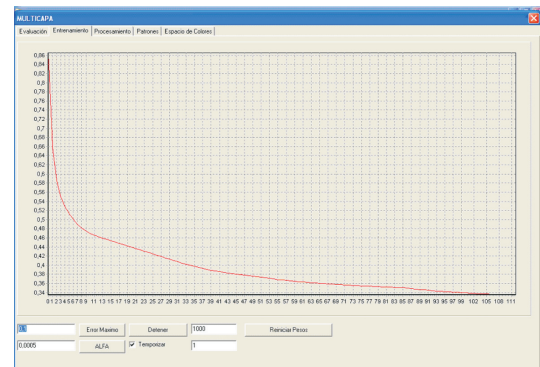


Figura 19. Grafica del error

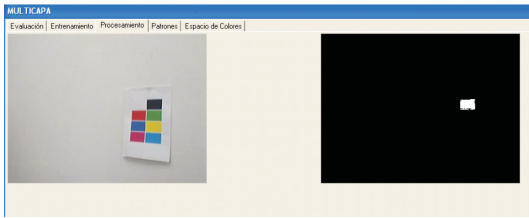


Figura 20. Imagen antes y después de usar el filtro entrenado

Luego de entrenar la red se realiza una medición de prueba para lo cual se ejecuta el programa Camaras.exe. Figura 20.

El resultado de la interpolación de las coordenadas se pueden ver gráficamente y analíticamente en las últimas secciones del programa, estas tienen el nombre de resultados gráficos y resultados numéricos. Figura 21.

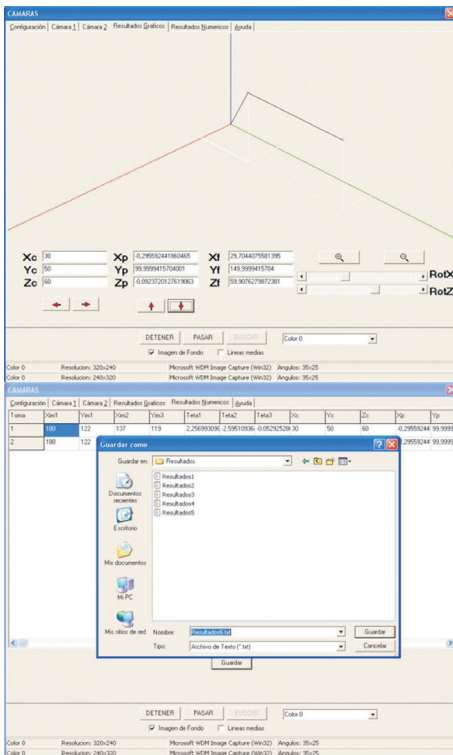


Figura 21. Resultado de Medición en forma gráfica y en forma numérica

IV. CONCLUSIONES

- Dada la relación directa entre píxeles y la medida del ángulo, es de suma importancia conocer la definición en píxeles y el lente que manejan cada una de las cámaras para minimizar ruido en la medición
- Dentro de la gama de filtros, el que mejor se comporta es el que utiliza el sistema HSV, con el cual no se tiene la dificultad de un valor único por color, se puede determinar un rango de valor en un canal para averiguar la tonalidad.
- Los colores de los objetos deben ser lo más puros posibles para evitar errores al pasar por los filtros de color (RBG o HSV).
- Las redes neuronales ofrecen una mejor alternativa a la solución de problemas complejos como lo es la visión artificial. Una red de 3 entradas y 3 salidas ofrece resultados satisfactorios (error menor al 2%).
- El algoritmo está dirigido por 3 parámetros: θ_1 , θ_2 (ángulos con respecto a la vertical de la cámara 1 y 2 respectivamente) y θ_3 (ángulo de elevación). De donde los ángulos θ_1 y θ_2 afectan con mayor peso la medida de la profundidad.
- El algoritmo se equivoca bastante en las zonas con pobre iluminación, por ejemplo bajo una mesa.

V. REFERENCIAS

[1] Carl Zeiss homepage, <http://www.zeiss.de/C12567A100537AB9/all-BySubject/BD477DE7D4E650C2C12569D8003C8166>.

[2] Del Brio Bonifacio M., Alfredo S. Molina; 2002. "Redes neuronales y sistemas difusos",

Madrid 2ª edición; Editorial Alfa-Omega; P. 63-64.

[3] Jang J. -S. R., C. -T. Su, E. Mizutani, 1998. "Neuro-Fuzzy and soft computing", United States. Prentice hall, P. 233-236

[4] Hart, Anna. Mar 1992. Using Neural Networks for Classification Tasks - Some Experiments on Datasets and Practical Advice. The Journal of the Operational Research Society Oxford.

[5] Danese Giovanni, Francesco Leporati, Stefano Ramat. May/Jun 2002. A parallel neural processor for real-time applications. IEEE Micro. Los Alamitos: Tomo 22, No. 3; pág. 20

[6] Geige C, R. Hunstock, G. Lehrenfeld, W. Mueller_, J. Quintanilla, C. Tahedl, A. Weber. Visual Modeling and 3D-Representation with a Complete Visual Programming Language —A Case Study inManufacturing_

[7] Duysinx, Pierre. Geradin, Michel. University of Liege, 2004. An Introduction to Robotics: Mechanical Aspects... p 5-21.

[8] Cyril Cauchois, Eric Brassart, Laurent Delahoche, Arnaud Clerentin. 3D Localization with Conical Vision
CREA (Center of Robotic, Electrotechnic and Automatic)

[9] S. Baker and S. K. Nayar. January 1998. "A Theory of Catadioptric Image Formation" Proceedings of the 6th International Conference on Computer Vision, Bombay, pp. 35-42.