



## Extracción de Líneas a partir de Escaneos Láser integrando Transformada de Hough, Mínimos Cuadrados Totales y Seguimiento de Bordes Sucesivos

*Lines Extraction from Laser Scans Integrating Hough Transform, Total Least Squares and Successive Edges Following*

J. Stephany Berrio P.  
Est. Maestría en Ingeniería  
Universidad del Valle  
julie.berrio@correounivalle.edu.co

Samith O. Ordoñez  
Est. Ingeniería Electrónica  
Universidad del Valle  
samith.orozco@correounivalle.edu.co

Eduardo F. Caicedo  
Profesor Titular  
Universidad del Valle  
eduardo.caicedo@correounivalle.edu.co

### Resumen

Este artículo presenta un método de segmentación de rectas, basado principalmente en un algoritmo de extracción de líneas robusto denominado transformada de Hough, para obtener los parámetros preliminares de las rectas compuestas por un conjunto de puntos suministrados por un escáner láser 2D, como los utilizados en sistemas de navegación de robots. Los parámetros de las rectas obtenidos por la transformada de Hough son ajustados por el algoritmo de Mínimos Cuadrados Totales (mejorando la precisión), y los puntos asociados a estos parámetros son nuevamente procesados por un algoritmo de Seguimiento de Bordes Sucesivos, con el fin de obtener las coordenadas cartesianas de los puntos extremos de las líneas que describen el entorno.

**Palabras clave:** rectas, segmentación, datos láser, coordenadas cartesianas, transformada Hough, mínimos cuadrados totales.

### Abstract

This paper presents a lines segmentation method, based mainly on a robust line extraction algorithm called Hough transform to obtain the preliminary parameters of straight lines composed of a set of points provided by a 2D laser scanner such as those used in robot navigation systems. The lines parameters obtained by the Hough transform algorithm are fitted by the Total Least Squares (improving the precision), and the points associated with these parameters are again processed by a Successive Edge Following algorithm in order to obtain the Cartesian coordinates of the extreme points of the lines which describe the environment.

**Key words:** straight lines, segmentation, laser data, Cartesian coordinates, Hough transform, total least squares.

### 1. Introducción

Una gran variedad de técnicas de localización y mapeo simultáneo se basan en las representaciones del ambiente de trabajo definidas por un grupo de características detectadas por el sistema sensorial de un robot (mapas basados en características) [1]. Para la represen-

Fecha recibido: Mayo 21 de 2012  
Fecha modificado: Junio 04 de 2012  
Fecha aceptado: Junio 08 de 2012



tación de puntos de referencia, el robot hace uso de los datos recogidos por el sensor, algunos algoritmos los representan como puntos en el espacio de trabajo, otros corresponden a una descripción de como la lecturas se agrupan (características geométricas). El anterior es uno de los más utilizados debido a la compacidad (menor capacidad de almacenamiento) sin pérdida significativa de información; esto incluye la extracción de líneas, curvas, esquinas, entre otros.

Entre las muchas formas geométricas, el segmento de línea es la más sencilla. Es fácil describir la mayoría de entornos semi-estructurados con rectas. Muchos algoritmos han sido propuestos para utilizar las características de las líneas extraídas de datos en 2D. Castellanos en [2], propone un método de segmentación de la línea que se inspira en un algoritmo de visión por computador, para ser utilizado con un mapa a priori como una aproximación a la localización del robot. Vandorpe en [3] presenta un algoritmo para la construcción de mapas dinámicos basados en las características geométricas (líneas y círculos) utilizando un escáner láser. Arras en [4] utiliza un escáner 2D con un método de segmentación basado en la regresión lineal. Jensfelt en [5] presenta una técnica para la adquisición y el seguimiento de la posición de un robot móvil con un escáner láser trazando líneas ortogonales (paredes) en un entorno de oficina. Por último, Pfister en [6] sugiere un algoritmo para la extracción de la línea de conexión ponderada para la construcción de mapas [7].

Por su parte, la Transformada de Hough es una herramienta que permite detectar curvas en un conjunto de lecturas láser o imágenes. Es una técnica muy robusta frente al ruido y a la existencia de brechas en la frontera del objeto, por ello es muy utilizada para aislar características de forma particular dentro de una imagen o escaneo láser. El objetivo de este artículo es encontrar curvas a partir de datos obtenidos por un láser 2D, que puedan ser parametrizadas como líneas rectas, este es un artículo extendido que fue presentado en el WEA 2012 [8]. La robustez del algoritmo a saltos en el contorno hace necesario la aplicación de una técnica que fraccione y obtenga puntos extremos de las rectas, para ajustar los resultados al entorno. Para esto se utilizó el algoritmo de Seguimiento de Bordes Sucesivos Consecutivos o SEF (*Successive Edge Following*) que considera la finalización de un segmento cuando la distancia entre las coordenadas de un punto y su consecutivo excede un determinado valor umbral fijado con antelación.

## 2. Definición del problema

En la problemática de Localización y Mapeo Simultáneo (SLAM), es importante precisar el tipo de características que describirán el mapa como son las líneas rectas; para obtener dichas marcas se recurre a datos sensoriales que brindan una apreciación del entorno. En el presente trabajo se utilizó un láser empotrado en un robot móvil. Los grupos de puntos obtenidos en el escaneo se procesan con algoritmos que permiten extraer líneas rectas y que son descritos en este apartado; al final del mismo se exponen los resultados obtenidos con la propuesta planteada.

### 2.1. Datos sensoriales

Un escaneo láser describe un corte 2D l ambiente de trabajo del robot, a través de una serie de puntos específicos en el sistema de coordenadas polares  $(r_R, \theta_R)$ , cuyo origen es la ubicación del sensor (Figura 1).

Se considera un robot navegando en un entorno 2D y se denota su ubicación  $(x_k = [x_r, y_r, \theta_r])$  (donde  $(x_r, y_r)$  y  $(\theta_r)$  son la posición y orientación respectivamente en determinado instante  $k$ , en un marco de referencia global  $(x_G, y_G)$ ). El espacio de trabajo es descrito por características estáticas rectilíneas (tales como partes de las paredes, puertas, estantes) detectados por el sistema sensorial del robot.

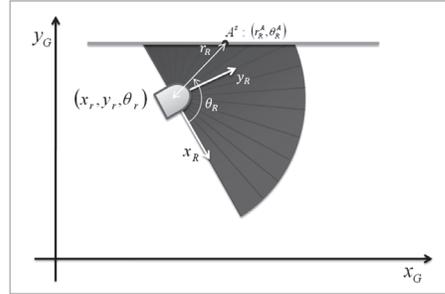


Figura 1. Parámetros del escáner láser

El robot está equipado con un sensor de proximidad (por ejemplo, los anillos de sonar o escáner láser) que proporcionan una serie de pasos  $S$  y medidas  $A^s = [r_r^A, \theta_r^A]$  que son parametrizados por su distancia  $r_r^A$  desde el origen hasta el punto en la dirección de detección  $\theta_r^A$ . La pareja puede considerarse como las coordenadas polares de la lectura en un robot centrado en un sistema de referencia. Las coordenadas cartesianas se denotan en la ecuación (1)

$$A^s = [x_R^A, y_R^A] = [r_r^A \cos \theta_r^A, y_r^A \sin \theta_r^A] \quad (1)$$

## 2.2. Transformada de Hough

Una línea recta en el plano de coordenadas  $(x-y)$  puede ser descrita completamente por solo dos parámetros, la pendiente y la intersección con eje  $y$ . La parametrización más familiar es:

$$y = m_o x + b_o \quad (2)$$

Donde  $m_o$  es la pendiente y  $b_o$  es la intersección con el eje  $y$ . Esta línea, cuando se dibuja en el plano de parámetros  $(m-b)$  se convierte en un punto. Esto se puede observar, cuando, dado un punto  $(x_o, y_o)$ , el conjunto de líneas a través de  $(x_o, y_o)$  (donde cada una es parametrizada por algún par  $(m, b)$  forman en una línea recta en el plano de parámetros  $(m-b)$  (Figura 2).

Ésta es evidentemente la ecuación de transformación que indica la relación lineal entre  $m$  y  $b$ :

$$b = -x_o m + y_o \quad (3)$$

Cuando los puntos que se encuentran alineados con otros en el plano de coordenadas  $(x-y)$ , se transforman en sus respectivas líneas en el plano de parámetros  $(m-b)$ , estas tendrán un único punto de intersección. Las coordenadas  $(m, b)$  del punto de corte son exactamente la pendiente y el punto de intersección con el eje  $y$  de la línea que describen los puntos. Usando esta parametrización, existen singularidades cuando la pendiente, la variable independiente en este caso, no tiene límite (cuando una línea en el plano de coordenadas es paralela al

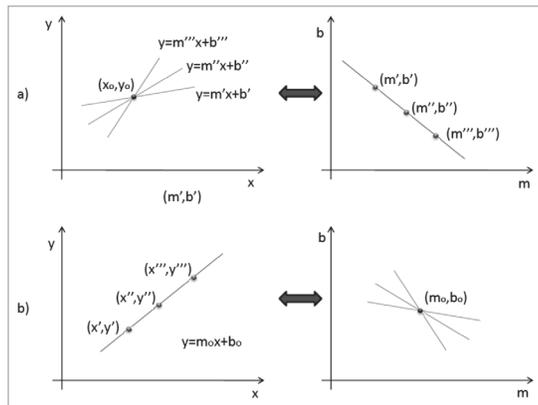


Figura 2. Transformaciones punto a línea. a) Un punto con líneas de varios parámetros  $(m, b)$  a través de estos se transforma en una línea recta en el plano de parámetros, b) Puntos colineales se transforman en líneas rectas que se interceptan en un único punto en el plano de parámetros.

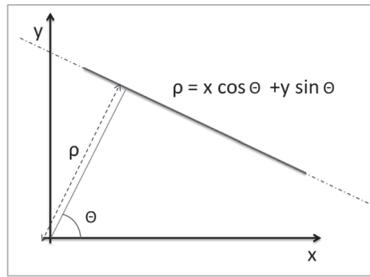


Figura 3. Parametrización normal.

eje  $y$ , la pendiente tiende a infinito). Como presenta Duda y Hart en [9], una manera más conveniente para describir la línea recta es utilizando la parametrización normal representada en la Ecuación (4) y Figura 3.

$$\rho = x \cos \theta + y \sin \theta \quad (4)$$

Una línea en el plano de coordenadas es descrita completamente por dos parámetros,  $\theta$  el ángulo de la normal desde el origen, y  $\rho$  su distancia del origen. Siguiendo la transformación punto línea, dado un punto  $(x_o, y_o)$  en el plano de coordenadas, el grupo de línea

a través de  $(x_o, y_o)$ , cada una de las cuales es parametrizada por algún par de  $(\theta, \rho)$ , será transformada en una curva sinusoidal en el plano de parámetros  $(\theta, \rho)$ , curva correspondiente a la Ecuación (4).

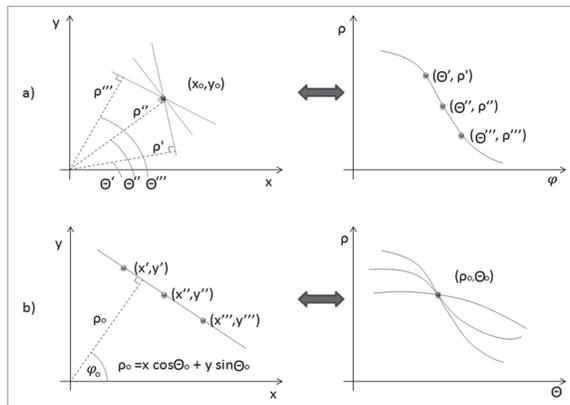


Figura 4. Transformaciones punto-curva. a) un punto con líneas con varios parámetros pasando a través de él se transforma en una curva sinusoidal en el plano de parámetros. b) Puntos colineales se transforman en curvas que se interceptan en un único punto en el plano de parámetros.

En este caso, los puntos se encuentran sobre una línea en el plano de coordenadas  $(x-y)$  son transformados en sus respectivas curvas sinusoidales en el plano de parámetros; estas curvas en el plano de parámetros se interceptarán en un único punto, las coordenadas  $(\theta, \rho)$  de este punto de intersección son exactamente el ángulo de la normal y la distancia desde el origen a la línea que pasa a través de los puntos en el plano de coordenadas. La Figura 4 ilustra las relaciones punto-curva [10]. La transformada de Hough (Algoritmo 1) hace uso de la transformación punto-curva; éste

algoritmo se basa en la premisa que puntos colineales en el plano de coordenadas serán transformadas en curvas con un único punto de intersección en el plano de parámetros, así, para encontrar una línea, el procedimiento es transformar los puntos a curvas en el plano de parámetros y localizar las intersecciones de las curvas para determinar la presencia de líneas.

Analizar el plano de parámetro para obtener intersecciones precisas sería una tarea tediosa e ineficiente. Por tal motivo se recurre a la discretización del espacio de parámetros, obteniendo una cuadrícula 2D donde cada celda representa una región en la cual las

**Algoritmo 1** Transformada Hough

**Entrada:**  $(x; y)$ , un conjunto de puntos en coordenadas cartesianas,  $Th\%$ , umbral de votación para identificación de líneas.

$\theta_d \leftarrow$  Vector de ángulos discreto de 0 a  $\pi$ .

**Para**  $i = 1$  hasta la longitud del conjunto de datos **hacer**

$$\rho(i) = x(i) \cos(\theta_d) + y(i) \sin(\theta_d)$$

**Fin para**

$\rho_d \leftarrow$  Parámetro  $\rho$  discretizado.

$im \leftarrow$  Cálculo de votación para cada pareja  $(\theta_d, \rho_d)$ .

$[r; c] \leftarrow$  Filas y columnas de  $im$  cuya votación es superior al umbral  $Th\%$ .

Extracción de los parámetros de las rectas  $(\rho_{im}, \theta_{im})$

intersecciones de las curvas podrían corresponder para ajustar los puntos colíndales. La resolución de la cuadrícula se determina por el grado de ajuste que deba tener la línea con los puntos.

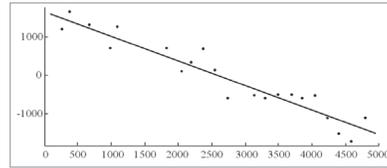


Figura 5. Recta descriptiva de los puntos de datos. Obtenida por Mínimos Cuadrados Totales.

### 2.3. Mínimos Cuadrados Totales

Difícilmente se podría nombrar a otro método que se utilice con tanta frecuencia como el método conocido como los Mínimos Cuadrados, aproximación que en estadística se considera como un estimador de máxima verosimilitud de un modelo de regresión lineal bajo criterios estándar (distribución normal residual de media cero y matriz de covarianza que es un múltiplo de la identidad). Del mismo modo, los Mínimos Cuadrados Totales o TLS (*Total Least Square*) es un estimador de máxima verosimilitud en el modelo de errores invariantes. El TLS fue desarrollado por Golub y Van Loan [11] como una técnica de solución para un sistema sobre determinado de ecuaciones y luego perfeccionado por Van Huffel [12], es una extensión del usual método de mínimos cuadrados, permitiendo manejar también ciertas inconsistencias en la matriz de datos. La popularidad del TLS es debida probablemente al hecho de que la minimización del residuo puede ser realizada cercanamente a la forma analítica; el método adicionalmente es lineal y es consistente con el principio de ortogonalidad de la aproximación óptima en sentido del error cuadrático medio.

TLS es un método de ajuste que se utiliza cuando hay errores tanto en el vector de observación como en la matriz de datos, por eso es una buena opción para obtener los parámetros apropiados de la línea final. El TLS minimiza la suma del cuadrado de las distancias de los puntos a la recta (medidas en perpendicular). En el caso del problema del TLS, se quiere tomar en cuenta las perturbaciones en la matriz de sensibilidad, para ello el TLS se define como la minimización sobre el rango y perturbaciones de la matriz de sensibilidad de tal forma que la distancia normal euclidiana entre la recta y los puntos que la componen sea la menor posible. En la Figura 5 se ilustra la recta resultante para un conjunto de datos arbitrarios.

En el presente trabajo, se agrupan los puntos más cercanos a la recta que ha sido detectada por la transformada Hough, estos son llamados *inliers*, y son usados para recalcular los parámetros de la recta que más se adecua a ellos como se detalla en el Algoritmo 2.

#### Algoritmo 2 TLS

Entrada:  $(x; y)$ , un conjunto de puntos en coordenadas cartesianas,  $\sigma$  la desviación estándar por cada punto.

$$x_{promedio} \leftarrow \frac{1}{n} \sum_{i=1}^n x_i$$

$$y_{promedio} \leftarrow \frac{1}{n} \sum_{i=1}^n y_i$$

$$dx \leftarrow x - x_{promedio}$$

$$dy \leftarrow y - y_{promedio}$$

$$N \leftarrow -2 \sum \frac{dx * dy}{\sigma^2}$$

$$D \leftarrow \frac{1}{\sigma^2} \sum (dy^2 - dx^2)$$

$$\theta \leftarrow \frac{atan(\frac{N}{D})}{2}$$

$$\rho \leftarrow x_{promedio} * \cos(\theta) + y_{promedio} * \sin(\theta)$$



Cabe anotar que los parámetros de la recta obtenidos a través de la aplicación del TLS están dados en forma polar paramétrica y de esta forma se ingresan los parámetros a un filtro extendido de Kalman para realizar el proceso de Localización y Mapeo Simultaneo (SLAM).

Cuando tenemos variables que son los valores de mediciones experimentales, estas tienen incertidumbres debido a limitaciones de medición (por ejemplo, la precisión del instrumento) que se propagan a la combinación de variables en la función. Para propagar la incertidumbre y obtener la matriz de covarianza de cada línea tenemos que  $(r_s, t_s)$  es un conjunto de puntos en coordenadas polares,  $(\sigma)$  la desviación estándar por cada punto, y  $(r_s \cos(t_s), r_s \text{sen}(t_s))$  su correspondiente en coordenadas cartesianas, como las ecuaciones para obtener los parámetros de la línea son un conjunto de combinaciones no-lineales de la variable  $r_s$ , estas deben ser linealizadas por aproximación a una expansión de primer orden en serie de Taylor, así:

$$\sigma_{\theta} = \frac{\partial \theta}{\partial r_s} * \sigma^2 \tag{5}$$

$$\sigma_{\rho} = \frac{\partial \rho}{\partial r_s} * \sigma^2 \tag{6}$$

$$\sigma_{\theta\rho} = \left( \left( \frac{\partial \theta}{\partial r_s} * \frac{\partial \rho}{\partial r_s} \right) * \sigma \right)^2 \tag{7}$$

### 2.4. Seguimiento de Bordes Sucesivos

Este algoritmo de Seguimiento de Bordes Sucesivos o SEF (*Successive Edge Following*) [13] [14] trabaja directamente con las distancias medidas por el escáner láser, que se encuentran en formato polar  $(\theta, \rho)$ . Por consiguiente, no se necesita llevar a cabo ninguna transformación a coordenadas cartesianas  $(x, y)$ . En esta técnica, presentada en el Algoritmo 3, se considera que un segmento termina cuando la diferencia entre una distancia y su siguiente  $(|\rho_{j+1} - \rho_j|)$  es mayor a un umbral (**Th%**) previamente fijado.

Al terminar el algoritmo se obtienen las coordenadas de los extremos de los segmentos de rectas que constituyen los puntos, así, se consiguen los conjuntos de líneas [15]. Un

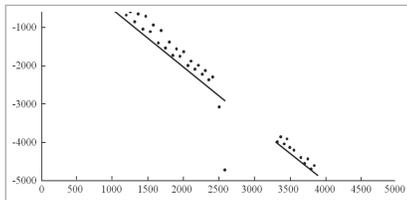


Figura 6. Corte de rectas según los puntos en Seguimiento de Bordes Sucesivos (SEF).

ejemplo del fraccionamiento de rectas realizado por el algoritmo cuando las distancias superan el umbral, se presenta en la Figura 6.

La principal ventaja de este algoritmo es su velocidad de cómputo, mas tiene el inconveniente de ser dependiente de un umbral que no varía y debe ser ajustado a los datos. Además, se pueden pro-

#### Algoritmo 3 SEF

```

Entrada:  $\rho_1 \dots, \rho_n$  un conjunto de medida de distancias
 $i \leftarrow 1$ 
Para  $j = 1$  hasta  $n - 1$  hacer
    Si  $\rho_{j+1} - \rho_j > Th\%$  entonces
         $S \leftarrow \{ \rho_i \dots, \rho_j \}$ 
        Almacenar  $S$ 
         $i = j + 1$ 
    Fin si
Fin para
    
```

ducir errores de detección en situaciones determinadas, como por ejemplo al posicionar el láser frente a un rincón (donde la diferencia entre las distancias consecutivas es poca), o una superficie donde se describa una línea recta con puntos muy separados entre sí.

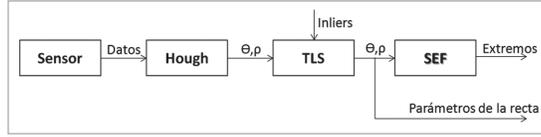


Figura 7. Esquema del algoritmo propuesto

## 2.5. Algoritmo Propuesto

El algoritmo propuesto, visualizado en la Figura 7, hace uso de tres técnicas de procesamiento: la Transformada de Hough (robusta), Mínimos Cuadrados Totales (robusta) y SEF (no robusta). El procedimiento completo se observa en el Algoritmo 4.

### Algoritmo 4 Algoritmo Propuesto

**Entrada:**  $(x; y)$ , un conjunto de puntos en coordenadas cartesianas,  $Tb$ , porcentaje de la máxima votación utilizado como umbral para identificación de líneas,  $Tb2$ , distancia máxima de una recta a un punto descrito por esta.

$\theta_d \leftarrow$  Vector de ángulos discreto de 0 a  $\pi$ .

// Transformada de Hough

**Para**  $i = 1$  hasta la longitud del conjunto de datos **hacer**

$$\rho(i) = x(i)\cos(\theta_d) + y(i)\sin(\theta_d)$$

**Fin para**

$\rho_d \leftarrow$  Parámetro  $\rho$  discretizado.

$im \leftarrow$  Cálculo de votación para cada pareja  $(\theta_d, \rho_d)$ .

$[r, c] \leftarrow$  Filas y columnas de  $im$  cuya votación es superior al umbral  $Tb$ .

Extracción de los parámetros de las rectas desde  $im(\rho_{0lin}, \theta_{0lin})$

// Ajuste de parámetros y obtención de extremos

**Para**  $k = 1$  hasta la cantidad de rectas encontradas **hacer**

Obtener puntos  $PtosIn_{0lin}$  cuya distancia a las líneas parametrizadas por  $(\rho_{0lin}, \theta_{0lin})_k$  sea menor a  $Tb2$ .

$PtosIn_{0lin}$  Son el conjunto de  $n$  puntos en coordenadas cartesianas  $(x, y)_{0lin}$ . Y  $\sigma$  la desviación estándar por cada punto.

$$x_{promedio} \leftarrow \frac{1}{n} \sum_{i=1}^n x_i$$

$$y_{promedio} \leftarrow \frac{1}{n} \sum_{i=1}^n y_i$$

$$dx \leftarrow x - x_{promedio}$$

$$dy \leftarrow y - y_{promedio}$$

$$N \leftarrow -2 \sum \frac{dx * dy}{\sigma^2}$$

$$D \leftarrow \frac{1}{\sigma^2} \sum (dy^2 - dx^2)$$

$$\theta_{Fin} \leftarrow \frac{atan(\frac{N}{D})}{2}$$

$$\rho_{Fin} \leftarrow x_{promedio} * \cos(\theta) + y_{promedio} * \sin(\theta)$$

Obtener puntos  $PtosIn_{Fin}$  cuya distancia a las líneas parametrizadas por  $(\rho_{Fin}, \theta_{Fin})$  sea menor a  $Tb2$ .

$i \leftarrow 1$

**Para**  $j = 1$  hasta el número de puntos clasificados dentro de las rectas **hacer**

**Si**  $PtosIn_{j+1} - PtosIn_j > Tb3$  **entonces**

$$S \leftarrow \{ PtosIn_j, \dots, PtosIn_{j+1} \}$$

Almacenar  $S$

$i \leftarrow j + 1$

**Fin si**

**Fin para**

**Fin para**



Primero se implementa la Transformada de Hough con el fin de obtener los parámetros de las rectas que mejor describen la nube de puntos (barrido láser), para esto se normalizan los valores de la matriz de votación de la cuadrícula discreta (para visualización), y se obtienen las coordenadas de aquellos valores mayores a un umbral. El umbral está dado como porcentaje del valor máximo de votación, asegurando de esta forma al menos una recta detectada por el algoritmo. Una vez se adquieren los parámetros de las líneas, se agrupan los puntos que más se ajustan a cada una de ellas, midiendo la distancia perpendicular de las rectas al punto y comparándola con un valor límite (que denota la pertenencia de un punto a una recta). Finalmente, el conjunto de datos es procesado por un algoritmo que los fracciona en grupos más pequeños dependiendo de la distancia existente entre estos (SEF), aquellos grupos que contienen menos de cinco datos son descartados. Con los restantes se hallan las coordenadas de inicio y fin de cada segmento de recta, que representa el ambiente escaneado.

## 2.6. Resultados

Para las pruebas del algoritmo se utilizaron datos obtenidos con un láser SICK que brinda 360 datos en un rango de 180°, ofreciendo una lectura cada 0,5°, para ello se escaneó consecutivamente un ambiente semiestructurado (fácilmente descrito por líneas rectas). Las experimentos fueron ejecutados en una máquina Dual Core 2.4Ghz AMD con MATLAB v7.10.

Las Figuras 8, 9 y 10 muestran algunos de los resultados experimentales que se obtuvieron con el algoritmo implementado, fijamos el parámetro  $Th\%$  en 40% para un primer escenario escaneado, el resultado del proceso lo encontramos en la Figura 8, la cual está compuesta de 4 gráficas, donde se muestran los puntos iniciales, la Transformada de Hough, la imagen de la cuadrícula de votación, y una última donde se presentan todas las rectas clasificadas en contraste con los puntos originales.

Para un segundo escaneo láser, se fija el parámetro  $Th\%$  en 50% y se ejecuta el algoritmo; la gráfica de los puntos y el proceso se hallan en la Figura 9, el resultado

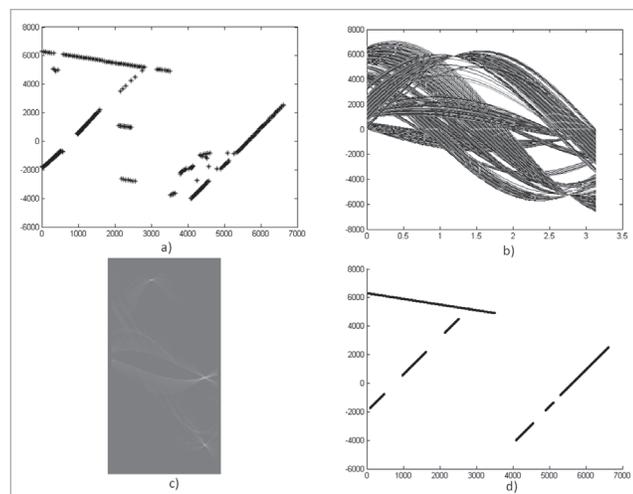


Figura 8. Resultados con umbral de 40% del voto máximo en la cuadrícula de la transformada de Hough. (a) Datos originales, (b) transformada de Hough de los puntos, (c) cuadrícula de votación 2D de la transformada de Hough, (d) Resultados del algoritmo.

corresponde a 2 líneas rectas detectadas, siendo una de ellas fraccionada en tres partes como consecuencia a los umbrales del SEF. Un tercer conjunto de datos es puesto a prueba, con un  $Th\%$  de 60%; en la Figura 10 se exponen 4 gráficas del proceso del algoritmo, para esta prueba igual que para la anterior, se extraen dos líneas, una de las cuales es dividida en varios segmentos.

Para una segunda prueba se establece un grupo de puntos que describen dos líneas que forman una esquina, para

cada prueba, el número de datos es modificado y procesado por el algoritmo, luego, para 100 datos se varía la cuadrícula desde un valor de 10.500 hasta 25.200 (el número de cuadrícula corresponde a la multiplicación las divisiones de  $\theta$  con  $\rho$ ), conservando el umbral  $Th\%$  en 80% y las divisiones  $\theta$  fijas en 105, variando solamente las divisiones de  $\rho$ . La Tabla I muestra cuanto tiempo en segundos se demora el algoritmo en procesar las líneas, y el error en relación con los parámetros reales.

En la Figura 11 se muestran los puntos y las líneas detectadas, para diferentes cantidades de datos describiendo el mismo entorno. En (b) se muestra solo una línea que clasifica con el valor umbral  $Th\%$  igual a 80%, esto es debido a los pocos datos, para tener parámetros más precisos  $Th\%$  se debe reducir o el número de cuadrícula ser mayor.

Para mejorar precisión, se considera la solución de implementar una cuadrícula mas fina, esto significa, que la cantidad de divisiones ( $\theta$  y  $\rho$ ) se incrementa, pero con esto también se incrementa el tiempo computacional, según la Tabla I. Dado el alto costo en tiempo computacional para adaptar la cuadrícula a las líneas con el fin de disminuir el error o simplemente aproximarse a los correctos parámetros de las rectas, entonces se adecua otra solución aplicando el método Mínimos Cuadrados Totales el cual no usa mucho tiempo y disminuye bastante el error. Lo anterior da como resultado una buena aproximación para la línea con una cuadrícula gruesa y tomando casi el mismo tiempo.

La columna de error  $m$  en la Tabla I corresponde a la diferencia entre el parámetro  $m$  obtenido y el valor de  $m$  real. De igual manera con la columna del error de el parámetro  $b$ . Como se observa en la Tabla I, el error es alto cuando no hay suficientes datos que caractericen la recta, la línea recta en el resultado es una línea ajustada a la cantidad de datos presentes y es por eso que el error también disminuye cuando se incrementa la cantidad de datos.

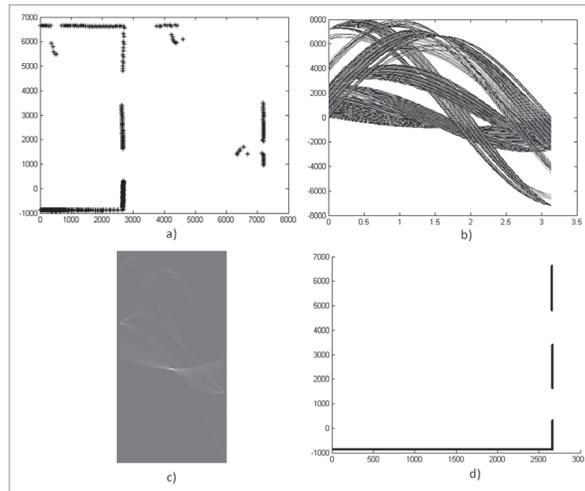


Figura 9. Resultados con umbral de 50% del voto máximo en la grilla de la transformada de Hough. (a) Datos originales, (b) transformada de Hough de los puntos, (c) grilla de votación 2D de la transformada de Hough, (d) Resultados del algoritmo.

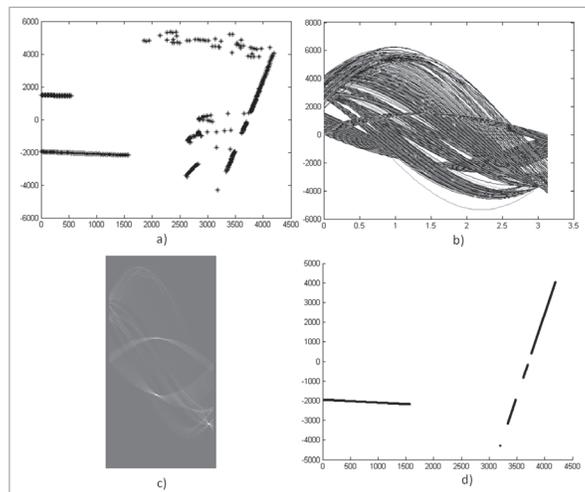
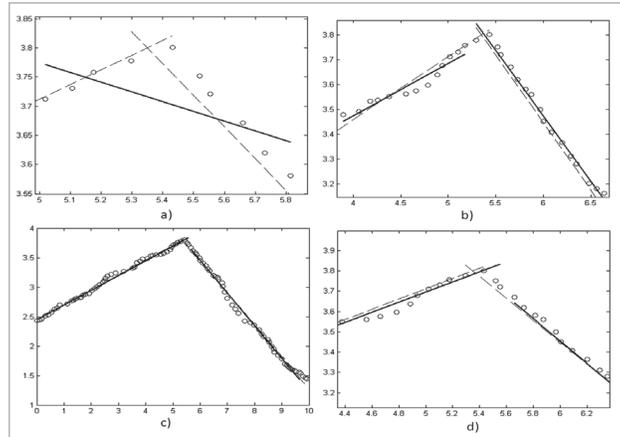


Figura 10. Resultados con umbral de 60% del voto máximo en la grilla de la transformada de Hough. (a) Datos originales, (b) transformada de Hough de los puntos, (c) grilla de votación 2D de la transformada de Hough, (d) Resultados del algoritmo.



**Figura 11.** Gráficas para el segundo método de validación. La línea punteada describe la esquina original (a) Resultado para 10 puntos. (b) Resultado para 30 puntos. (c) Resultado para 100 puntos. (d) Acercamiento a la esquina del resultado para 100 puntos. Para todas las pruebas se usó una cuadrícula de 10.500 divisiones.

**Tabla I.** Prueba de error de los puntos de datos contra líneas reales cuando se incrementan las divisiones de la grilla.

Datos	Grilla	Tiempo(s)	m1	b1	m2	b2	error m	error b
10	10.500	0,0242	-0,1675	4,6121	-0,1675	4,6121	0,3980	2,1290
20	10.500	0,0249	0,3442	1,9728	-0,5276	6,6402	0,0532	0,2690
30	10.500	0,0248	0,2104	2,6332	-0,5276	6,6397	0,0290	0,1257
40	10.500	0,0256	0,2276	2,5526	-0,5397	6,7102	0,0144	0,0559
50	10.500	0,0259	0,2135	2,6116	-0,6219	7,2164	0,0593	0,3385
60	10.500	0,0260	0,2290	2,5398	-0,5835	6,9724	0,0323	0,1807
70	10.500	0,0266	0,2497	2,4636	-0,5664	6,8602	0,0135	0,0865
80	10.500	0,0274	0,2445	2,4711	-0,5646	6,8529	0,0151	0,0865
90	10.500	0,0277	0,2508	2,4553	-0,5561	6,8007	0,0078	0,0526
100	10.500	0,0282	0,24918	2,4497	-0,55399	6,7761	0,0075	0,0375
100	16.800	0,0433	0,25808	2,4393	-0,56094	6,8426	0,0115	0,0727
100	21.000	0,0534	0,25357	2,4422	-0,56094	6,8426	0,0093	0,0713
100	25.200	0,0687	0,25365	2,4455	-0,56094	6,8426	0,0093	0,0696
100	31.500	0,0971	0,25365	2,4455	-0,56094	6,8426	0,0093	0,0696
100	42.000	0,1311	0,25480	2,4438	-0,56094	6,8426	0,0099	0,0705
100	52.500	0,1634	0,25365	2,4455	-0,56094	6,8426	0,0093	0,0696
100	105.000	0,3234	0,25318	2,4463	-0,56177	6,8327	0,0095	0,0642
100	126.000	0,3878	0,25318	2,4463	-0,55961	6,8211	0,0084	0,0585
100	147.000	0,4541	0,25391	2,4450	-0,55770	6,8084	0,0078	0,0527
100	189.000	0,5783	0,25375	2,4419	-0,55770	6,8084	0,0077	0,0543

En la última prueba se realiza con 180 datos de un escaneo láser (ilustrado en la Figura 12), a los cuales se le aplica el algoritmo propuesto, evaluando el ajuste de parámetros con Mínimos Cuadrados Totales con los puntos *inliers* generados en la detección de líneas con la transformada de Hough. Se realizan pruebas de tiempo con y sin TLS variando las divisiones de  $\rho$  desde 3 hasta 45, con un umbral  $Th\%$  de 50%.

Las Figuras 13 y 14 contienen los resultados del algoritmo para 6, 12, 21 y 24 divisiones de  $\rho$ , donde se aprecia el ajuste de TLS y la aproximación obtenida por la transformada de Hough, con un  $\rho$  fijo de 50%. En la Figura 15 se grafica el resultado de la misma prueba, pero con 30 y 45 divisiones en  $\rho$ . Nótese que el algoritmo que contiene el TLS arroja los mismos resultados, aunque se incremente la cuadrícula; sin TLS, la detección de los parámetros resulta menos exacta.

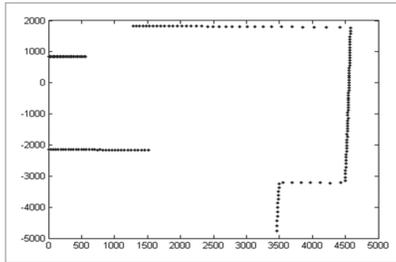


Figura 12. Escaneo láser de 180 puntos.

### 3. Conclusiones

Los umbrales que limitan distancias, tanto de inclusión de un punto a una línea, como la de separación entre puntos, están relacionados directamente con el tamaño del espacio de datos. Para conservar la coherencia entre líneas y puntos, en el algoritmo de SEF, si el valor umbral es bajo, la línea es fraccionada en varios segmentos.

El parámetro umbral de votación para la extracción de líneas ( $Th$ ) fue determinado como un porcentaje del valor máximo de la cuadrícula de votación, lo que es útil cuando no se conoce con precisión el número de puntos a ser procesados (el voto máximo de la cuadrícula 2D depende de la cantidad de datos), ya que para el caso de un escaneo láser, algunos datos son descartados por estar fuera del rango de detección del sensor.

Por su parte, se encontró que el método de TLS ajusta parámetros de baja precisión, en un tiempo computacional reducido en comparación a la solución de incrementar la cuadrícula para mejorar los resultados. La integración de las técnicas (Transformada Hough-robusto, TLS-robusto, SEF-no robusto), no afecta significativamente el tiempo de procesamiento ya que los dos últimos se caracterizan por su

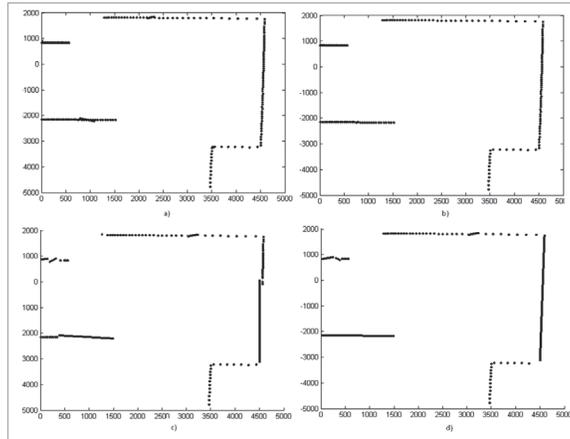


Figura 13. Gráficas de la prueba de ajuste con TLS. a) 6 divisiones para  $\rho$  sin TLS, b) 6 divisiones para  $\rho$  con TLS, c) 12 divisiones para  $\rho$  sin TLS, d) 12 divisiones para  $\rho$  con TLS.

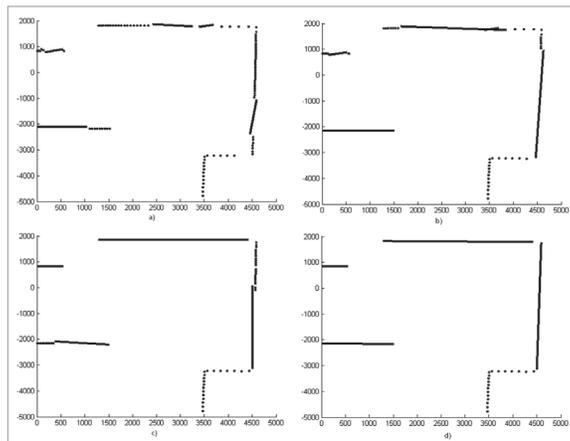


Figura 14. Gráficas de la prueba de ajuste con TLS. a) 21 divisiones para  $\rho$  sin TLS, b) 21 divisiones para  $\rho$  con TLS, c) 24 divisiones para  $\rho$  sin TLS, d) 24 divisiones para  $\rho$  con TLS.

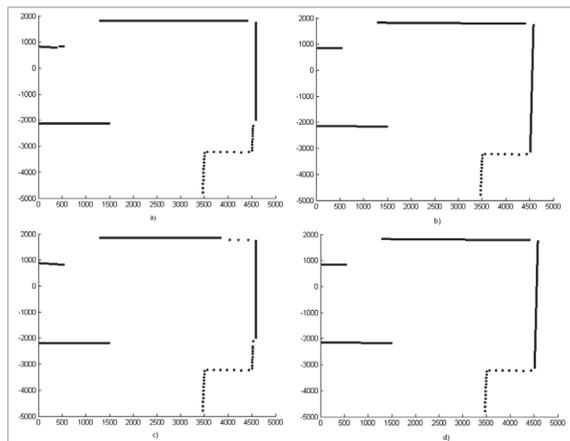


Figura 15. Gráficas de la prueba de ajuste con TLS. a) 30 divisiones para  $\rho$  sin TLS, b) 30 divisiones para  $\rho$  con TLS, c) 45 divisiones para  $\rho$  sin TLS, d) 45 divisiones para  $\rho$  con TLS.



velocidad, que es ventajoso para mejorar los resultados en tiempo menor que el necesitado por la transformada de Hough para obtener resultados similares.

Basados en los resultados de la investigación, el trabajo futuro incluye tanto la aplicación de los algoritmos en técnicas de mapeo y localización simultánea para un robot móvil, como su implementación en hardware con el fin de mostrar y evaluar su desempeño tiempo real.

## Referencias

- [1] A. Garulli, A. Giannitrapani, A. Rossi y A. Vicino, "Mobile robot SLAM for line-based environment representation". *European Control Conference 2005*, pp. 2041-2046, 2005.
- [2] J. A. Castellanos y J. D. Tardós, "Laser-based segmentation and localization for a mobile robot". *Robotics and Manufacturing vol. 6 - Procs. of the 6th Int. symposium (ISRAM)*, p. 101-108, 1996.
- [3] H. X. J. Vandorpe y H. Van Brussel, "Exact dynamic map building for a mobile robot using geometrical primitives produced by a 2d range finder". *In Proceedings of the IEEE International Conference n Robotics and Automation*, p. 901-908., 1996.
- [4] K. O. Arras y R. Siegwart, "Feature Extraction and Scene Interpretation for Map-Based Navigation". *Symposium on Intelligent Systems and Advanced Manufacturing*, 1997.
- [5] D. Kragic, P. Jensfelt, J. Folkesson y M. Bjorkman, "A framework for vision based bearing only {3D} {SLAM}". *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'06)*, 2006.
- [6] S. T. Pfister, S. I. Roumeliotis y J. W. Burdick, "Weighted line fitting algorithms for mobile robot map building and efficient data representation". *In ICRA*, p. 14-19., 2003.
- [7] V. Nguyen, S. Gächter, M. Agostino, N. Tomatis y R. Siegwart, "A comparison of line extraction algorithms using 2D range data for indoor mobile robotics" *Auton. Robots*, vol. 23, nº 15, pp. 97-111, 2007.
- [8] J. Berrío, S. Orozco, E. Caicedo, "Lines extraction in laser scans through the integration of the Hough transform and SEF", *2012 Workshop on Engineering Applications*, 2012.
- [9] R. O. Duda y P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures". *Commun. ACM*, vol. 15, nº 5, pp. 11-15, 1972.
- [10] X. Yun, K. Latt y J. Glennon, "Mobile robot localization using the Hough transform and neural networks". *Intelligent Control (ISIC), 1998. Held jointly with IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA), Intelligent Systems and Semiotics (ISAS), Proceedings*, pp. 393-400, 1998.
- [11] G. H. Golub y C. Van Loan, "An Analysis of the Total Least Squares Problem", Ithaca, NY, USA: Cornell University, 1980.
- [12] J. Vandewalle y S. Van Huffel, "Analysis and solution of the nongeneric total least squares problem". *SIAM J. Matrix Anal. Appl.*, vol. 9, nº 13, pp. 360-372, 1988.
- [13] M. Aldon, G. Borges y J., "A split-and-merge segmentation algorithm for line extraction in 2D range images". *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, pp. 441-444, 2000.
- [14] A. Siadat, A. Kaske, S. KLAUSMANN, D. Michel y R. HUSSON, "An Optimized Segmentation Method For A 2D Laser-Scanner Applied To Mobile Robot Navigation". *3rd IFAC symposium on Intelligent Components and Instruments for Control Applications*, 1997.
- [15] C. F. Caramés, "Técnicas de navegación de robots basados en sistemas de medición por láser", Universidad de Salamanca, 2007.

### Julie Stephany Berrío Perez

Ingeniera Mecatrónica de la Universidad Autónoma de Occidente, de Cali, Colombia. Obtuvo su título de Maestría en Ingeniería con énfasis en Ingeniería Electrónica en la Universidad del Valle de Cali, Colombia. Se desempeñó como ingeniera de proyectos en RT Solutions Ltda durante 1 año. Posteriormente, ejerció el cargo de asistente de docencia en la Universidad del Valle donde desarrolló el proyecto: Mapeo y localización simultánea de un robot móvil en ambientes estructurados basado en integración sensorial. Actualmente pertenece como investigador al grupo de percepción y sistemas inteligentes de la universidad del valle donde realiza estudios sobre robótica móvil.

### Samith Aron Orozco Ordoñez

Estudiante de Ingeniería Electrónica en la Universidad del Valle, de Cali, Colombia. Actualmente pertenece como investigador al grupo de percepción y sistemas inteligentes de la universidad del valle donde realiza estudios sobre robótica móvil.

### Eduardo Francisco Caicedo Bravo

Ingeniero Electricista de la Universidad del Valle, de Cali, Colombia. Obtuvo su título de Maestría en Tecnologías de la Información en Fabricación en la Universidad Politécnica De Madrid, España. PhD en Informática Industrial en la Universidad Politécnica De Madrid, España. Se desempeñó como director de nuevas tecnologías y educación virtual, e hizo parte del Consejo Superior de Investigaciones Científicas en la Universidad del Valle de Cali, Colombia. Actualmente se desempeña como profesor titular en el área de electrónica en la Universidad del Valle de Cali, Colombia, y es el director del grupo de investigación de percepción y sistemas inteligentes donde realiza estudios sobre inteligencia computacional, arquitectura y concepción de sistemas inteligentes, procesamiento digital de señales e imágenes, percepción artificial y robótica avanzada.