

UNA APROXIMACIÓN AL PROBLEMA DE LA DETECCIÓN DE ELIPSES EN PRESENCIA DE RUIDO Y VALORES ATÍPICOS

Alberto Ochoa*, Yasel Garcés**, Manuel Alejandro**

*Instituto de Cibernética, Matemática y Física.

**Universidad de la Habana.

ABSTRACT

We propose a new approach for ellipses detection in scattered data. The method is constructed on the assumption that it is possible to estimate the tangents to the ellipse in given points. From this information the coefficients of the quadratic form of the ellipses can be recovered. On this base two univariate cost functions are defined. The minimization of the proposed functions produces robust results in the presence of outliers, something difficult to obtain with least squares methods. With the aim of reducing the computational cost, one of the proposed functions approximates the intended ellipse with a polygonal produced by an efficient subdivision scheme.

KEY-WORDS : ellipse fitting, ellipse detection, subdivision schemes.

MSC: 68T20, 68W20, 62H12, 62H15, 62H20

RESUMEN

Se propone un nuevo enfoque para el problema de ajuste y detección de elipses a partir de datos dispersos. El método se construye sobre el supuesto de que es posible estimar las tangentes a la elipse en puntos dados. A partir de esta información es posible recuperar los coeficientes de la forma cuadrática de la cónica. Sobre esta base se definen dos funciones de costo univariadas, cuya optimización produce resultados robustos en presencia de valores atípicos, algo difícil de lograr con los enfoque mínimos cuadrados. Con la intención de reducir el costo computacional una de las funciones propuestas aproxima la elipse con una poligonal producida por un eficiente esquema de subdivisión.

1. INTRODUCCIÓN

Se propone un acercamiento al problema de detección y ajuste de elipses a datos dispersos, lo cual constituye un tema de gran importancia en aplicaciones de reconocimiento de patrones visuales [2, 4, 8, 9, 10, 12, 13, 15, 16, 18, 20, 21].

Este problema ha sido investigado utilizando diferentes técnicas como por ejemplo la transformada de Hough [25, 26, 27], el método RANSAC [19, 24], los filtros de Kalman [17, 22], el agrupamiento difuso [3, 11], los algoritmos evolutivos [15], o las aproximaciones por mínimos cuadrados [2, 10, 23]. En este contexto se distinguen dos grandes grupos: métodos de agrupamiento/votación y técnicas de optimización.

Los métodos de agrupamiento/votación tienen la ventaja de poder detectar varias elipses al mismo tiempo y ser robustos en presencia de valores atípicos (puntos que no pertenecen a las elipses). El precio a pagar es alto:

*ygarces@icimaf.cu

imprecisiones y alto consumo de tiempo y memoria. Dentro del grupo de métodos de optimización se distinguen los basados en mínimos cuadrados y los basados en metaheurísticas, en particular los algoritmos evolutivos. La gran flexibilidad de estos últimos les permite enfrentar la mayoría de los desafíos del problema de detección/ajuste de elipses. Su punto más débil es la eficiencia. Los métodos basados en mínimos cuadrados son muy rápidos y exactos pero tienen alta sensibilidad a los valores atípicos, lo que entre otras cosas, les impide trabajar con más de una elipse a la vez.

En este trabajo se estudian nuevas alternativas de solución a este problema y se comparan con el método de mínimos cuadrados. Las propuestas perdiendo un poco de eficiencia logran un comportamiento más robustos ante valores atípicos.

El reporte está organizado de la siguiente manera. En la sección 2 se introduce una de las propuestas del trabajo: una función de costo para el problema de ajuste de elipses basada en la obtención de los parámetros de la primitiva a partir de la estimación de tangentes. En la sección 3 se introduce otra función, esta vez utilizando un esquema de subdivisión interpolatorio que aproxima la elipse con una poligonal. La sección 4 hace una breve y muy preliminar incursión al problema de la estimación de tangentes en datos dispersos. Le sigue la sección de experimentos y luego las conclusiones del trabajo.

2. AJUSTE DE ELIPSES BASADO EN ESTIMACIÓN DE TANGENTES

El algoritmo DLSF (“direct least square fitting”) propuesto por Fitzgibbon et al. [9] es la columna vertebral de los algoritmos más eficientes existentes para la detección y ajuste de elipses. No obstante, este método presenta problemas cuando el ajuste se realiza en presencia de valores atípicos [12].

DLSF asume que se tiene un conjunto de puntos del plano y que estos pertenecen a una sola elipse, quizás con la presencia de cierta cantidad de ruido. En este trabajo se asume lo mismo, en el entendido de que es posible estimar las tangentes a la elipse en dos puntos diferentes de la misma. En presencia de ruido y valores atípicos, esto significa que es razonablemente posible seleccionar dos puntos no atípicos y diferentes de la elipse localizados en la vecindad de su perímetro.

La idea de estimar las tangentes a la elipse en un par de puntos tiene como objetivo extraer información de los datos que permita reducir la sensibilidad del método ante la presencia de valores atípicos.

Nótese que existen aplicaciones donde se quiere obtener la elipse que mejor describe los datos con independencia de su forma real. Este caso no es de interés para el presente trabajo.

El método que se introduce en esta sección es iterativo pero eficiente. Es decir, si lo comparamos con el método DLSF, podemos decir que se entrega un poco de eficiencia a favor de robustez. La buena noticia es que la función de costo que se utiliza para minimizar los errores entre los datos y la elipse de ajuste es univariada. Contraste esto con la optimización de seis variables que implícitamente realiza el sistema de valores propios del método DLSF.

En la siguiente sección mostraremos que los parámetros de la elipse de ajuste se pueden conocer a partir de la información de las tangentes en dos puntos dados. A partir de estos coeficientes, se calcula toda la información que se necesita sobre la curva. Por ejemplo, para la función de costo que se presenta en la Sección (2.2) es suficiente obtener el tamaño de los ejes a_e y b_e .

2.1. Forma general de la cónica dadas las tangentes

La forma cuadrática

$$f(x, y) = ax^2 + bxy + cy^2 + dx + ey + f$$

representa una cónica, que para $b^2 - 4ac < 0$ es una elipse.

Sean $A = (x_A, y_A)$ y $C = (x_C, y_C)$ puntos que pertenecen a una sección cónica y t_A, t_C las tangentes en A y C , respectivamente. En esta sección derivaremos una representación alternativa de una sección cónica:

$$(a, b, c, d, e, f) = \mathcal{F}(A, t_A, C, t_C, \omega). \quad (1)$$

La expresión

$$c(t) = \frac{b_0 B_0^2(t) + \omega b_1 B_1^2(t) + b_2 B_2^2(t)}{B_0^2(t) + \omega B_1^2(t) + B_2^2(t)}, \quad (2)$$

es la representación de una cónica en la forma racional de Bernstein-Bézier.

$$B_i^2(t) = \binom{2}{i} t^i (1-t)^{2-i}, \quad t \in [0, 1]$$

son los polinomios de Bernstein de grado 2, y ω es el parámetro de tensión local que controla la forma de la curva. La curva es una elipse si $0 < \omega < 1$, una parábola cuando $\omega = 1$ y una hipérbola para $\omega > 1$. Si ω define a la sección cónica $c(t)$, entonces con $-\omega$ se obtiene la ecuación de la sección cónica complementaria, denotada por $\hat{c}(t)$ [7].

Sea $B = (x_B, y_B)$ punto de intersección de las rectas tangentes t_A y t_C . De esta forma queda definido el triángulo de control $\{A, B, C\}$ que inscribe la sección cónica.

En el interior del triángulo de control la sección cónica se puede representar en coordenadas baricéntricas [7]

$$\beta^2 - 4\omega^2 \alpha(1 - \alpha - \beta) = 0. \quad (3)$$

La relación entre coordenadas cartesianas y baricéntricas de los puntos en el interior del triángulo de control está dada por

$$x = \alpha x_A + \beta x_B + \gamma x_C, \quad y = \alpha y_A + \beta y_B + \gamma y_C, \quad (4)$$

con $0 \leq \alpha, \beta, \gamma \leq 1$ y $\gamma = 1 - \alpha - \beta$.

Considerando la ecuación (4) para x , se puede expresar β en función de α y recíprocamente α en función de β . Obteniéndose

$$\alpha = \frac{x + \beta(x_C - x_B) - x_C}{x_A - x_C}, \quad \beta = \frac{x + \alpha(x_C - x_A) - x_C}{x_B - x_C}. \quad (5)$$

De igual forma, pero considerando la ecuación (4) para y se obtiene,

$$\alpha = \frac{y + \beta(y_C - y_B) - y_C}{y_A - y_C}, \quad \beta = \frac{y + \alpha(y_C - y_A) - y_C}{y_B - y_C}. \quad (6)$$

Combinando (5) y (6) se obtiene

$$\begin{aligned} \hat{\alpha} &= \frac{x(y_b - y_c) + y(x_c - x_b) + y_c x_b - y_b x_c}{(y_b - y_c)(x_a - x_c) - (y_a - y_c)(x_b - x_c)} \\ \hat{\beta} &= \frac{x(y_a - y_c) + y(x_c - x_a) + y_c x_a - y_a x_c}{(y_a - y_c)(x_b - x_c) - (y_b - y_c)(x_a - x_c)} \end{aligned}$$

Finalmente, sustituyendo $\hat{\alpha}$ y $\hat{\beta}$ en (3) y agrupando los términos semejantes se obtienen los coeficientes de la forma general de la cónica.

2.2. Función de costo basada en Hipérbolas Cofocales

La función de costo se define utilizando resultados publicados en [20].

Las familias de elipses e hipérbolas que son cofocales son mutuamente ortogonales en la proximidad de sus intersecciones. Debido a esto, la hipérbola cofocal que pasa a través de P puede ser una aproximación razonable a la línea recta que pasa por P , que es normal a la elipse.

En lo adelante se asume que el eje mayor de la elipse de ajuste es paralelo al eje x . La extensión de la derivación que sigue para elipses rotadas es simple.

El procedimiento primero determina la única hipérbola que pasa a través del punto P y es cofocal a la elipse dada por (1). Luego se calculan los cuatro puntos de intersección entre las curvas. Finalmente, se estima la distancia desde P a la elipse por la menor distancia desde éste a los cuatro puntos de intersección.

La posición de los focos de la elipse sobre el eje x se definen como $f_e = \pm\sqrt{a_e^2 - b_e^2}$. Para la hipérbola se tiene

$$f_h = \pm\sqrt{a_h^2 + b_h^2}, \quad (7)$$

donde a_e, b_e y a_h, b_h son los semiejes mayor y menor de la elipse y de la hipérbola.

Como solo se consideran las cónicas cofocales, entonces $f_e = f_h$. Denotando

$$A = a_h^2, \quad F = f_h^2 = f_e^2, \quad X = px^2, \quad Y = py^2, \quad (8)$$

y sustituyendo (8) en la ecuación implícita de la hipérbola se tiene

$$\frac{X}{A} - \frac{Y}{F - A} = 1. \quad (9)$$

Despejando la ecuación (9) en función de A se obtiene una forma cuadrática en A ,

$$A^2 - A(X + Y + F) + FX = 0. \quad (10)$$

Resolviendo (10) y sustituyendo en (8) y en (7), se obtienen a_h y b_h . Los puntos de intersección $I = (x_i, y_i)$ se hallan mediante las ecuaciones implícitas de la elipse y la hipérbola. Los 4 puntos de intersección quedan determinados por las siguientes relaciones

$$\left(x_i = \pm\sqrt{\frac{a_e^2(b_e^2 + b_h^2)}{a_h^2 b_e^2 + a_e^2 b_h^2}}, \quad y_i = \pm\frac{b_e b_h \sqrt{a_e^2 - a_h^2}}{\sqrt{a_h^2 b_e^2 + a_e^2 b_h^2}} \right). \quad (11)$$

Para obtener la aproximación de la distancia del punto P a la elipse se considera la distancia mínima d_P respecto a los cuatro puntos. Para más detalles consultar [20]. La función de costo queda definida de manera general como

$$\mathcal{H}(S_p, \xi) = \sum_{P \in S_p} d_P^2,$$

donde S_p denota al conjunto dato y ξ la elipse de ajuste.

Dado (A, t_A, C, t_C, ω) , los componentes más importantes del costo de evaluación de la función \mathcal{H} son: cálculo de los coeficientes de la forma cuadrática, cálculo de los semiejes y ángulo de rotación de la elipse a partir de los coeficientes y finalmente cálculo de los segmentos de las hipérbolas. En un algoritmo iterativo, estos costos se repetirán para los diferentes valores de ω que visite el optimizador. Teniendo en cuenta esto, en la próxima sección se introduce otra función de costo que no calcula ni los coeficientes de la forma cuadrática ni ningún parámetro de la elipse. El método se basa en un esquema de subdivisión que genera los puntos de las secciones cónicas dados los datos (A, t_A, C, t_C, ω) .

El algoritmo basado en la función de costo descrita en este acápite lo denominamos "Orthogonal Hyperbolae Fitting" (OHF).

3. UNA FUNCIÓN DE COSTO BASADA EN UN ESQUEMA DE SUBDIVISIÓN

Un proceso de subdivisión univariado define una curva como límite de una secuencia de poligonales que son refinamientos de una poligonal inicial. En esta sección se define una función de costo que estima la distancia de un punto $P_i \in Sp$ a la elipse de ajuste ξ , como la distancia del punto a la poligonal de control $\mathfrak{S}(\xi)$ generada mediante el proceso de subdivisión.

En la próxima sección se hace una breve introducción al método de subdivisión que se emplea en este trabajo. La exposición es parcial y muy breve; el lector interesado encontrará más detalles en la literatura citada.

3.1. Esquema de Subdivisión Interpolatorio

Si los puntos generados en cada etapa del refinamiento se mantienen durante todo el proceso, el esquema se dice interpolante [5].

El algoritmo de refinamiento (subdivisión) de la poligonal que tiende a la cónica que se desea aproximar consiste en insertar entre cada par de vértices consecutivos el punto con coordenadas baricéntricas $\left(\frac{1}{2(1+\omega)}, \frac{\omega}{1+\omega}, \frac{1}{2(1+\omega)}\right)$ [5, 6, 7]. Este se obtiene al evaluar la representación de la cónica en la forma racional de Bernstein-Bézier para $t = \frac{1}{2}$, lo cual es muy eficiente. Esta elección tiene la ventaja adicional de que no es necesario estimar la tangente en este punto ya que es paralela a $\overline{P_1P_2}$ [6]. Este punto se conoce con el nombre *shoulder point* (*sp*). El algoritmo realiza tantas iteraciones como se desee, tomando en lugar de la pareja P_1, P_2 , primero a la pareja P_1, sp y luego a P_2, sp y así sucesivamente.

A medida que se avanza en el refinamiento de la subdivisión, los triángulos de control respecto a los segmentos de arcos correspondientes van cambiando. Aún en los casos en que se trata de generar el mismo tipo de cónica, es necesario definir cuál será el nuevo parámetro que la identifica en la forma racional respecto a los nuevos triángulos [5].

Teorema 1 *Sea ω_j^i el parámetro de tensión local asociado al arco de cónica que interpola la arista i -ésima en la iteración j -ésima, entonces se cumple que*

$$\omega_{j+1}^{2i-1} = \omega_{j+1}^{2i} = \sqrt{\frac{1 + \omega_j^i}{2}}.$$

La demostración se encuentra disponible en [6].

De esta manera, el cómputo de ω para cada paso en la subdivisión se obtiene de manera recurrente y muy sencilla, lo cual es muy eficiente computacionalmente.

En general con pocas iteraciones del algoritmo de subdivisión se obtiene una buena aproximación de la cónica.

3.2. Distancia a una Sección Cónica

El objetivo de esta función de costo se basa en asociar cada punto del conjunto Sp a un segmento de la poligonal que se genera mediante el proceso de subdivisión descrito en la sección anterior. Hecho este particionamiento la distancia del conjunto de puntos Sp a la elipse de ajuste se aproxima mediante la suma de las distancias de cada punto de las particiones al segmento de la poligonal asociado.

Consideremos los puntos p_1, p_{2^k+1} que pertenecen a la elipse, y p^* punto medio del segmento $\overline{p_1p_{2^k+1}}$. Mediante el proceso de subdivisión en k iteraciones se obtienen $\{p_i, i = 2, 3, \dots, 2^k\}$ nuevos puntos, que junto a p_1 y p_{2^k+1} constituyen los vértices de la poligonal que aproxima a la cónica. De esta forma si se consideran las rectas $r_i, i = 1, 2, \dots, 2^k + 1$ que se forman mediante la unión de los puntos p_i con p^* se tiene una partición natural del conjunto Sp en 2^k subconjuntos de la forma

$$\widehat{Sp}_i = \{p \in Sp \mid p \text{ esta entre las rectas } r_i \text{ y } r_{i+1}\}$$

donde \widehat{Sp}_i queda asociado al segmento de poligonal $\overline{p_i, p_{i+1}}$. La siguiente figura muestra la partición del conjunto Sp para el caso de dos pasos en el algoritmo de subdivisión.

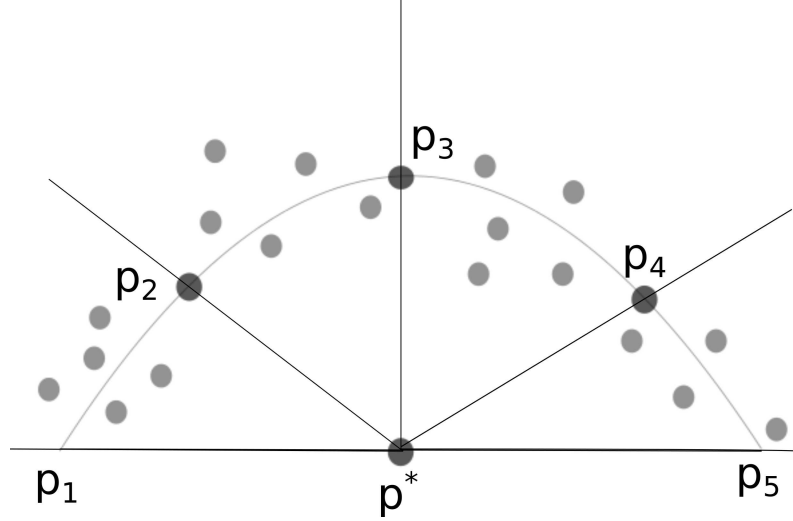


Figura 1: Funcionamiento del proceso de partición para $k = 2$.

La distancia de cada punto de \widehat{Sp}_i al segmento asociado es de fácil computo y permite no solo una buena aproximación de la distancia real a la elipse sino que también puede actuar como un filtro para eliminar puntos outliers.

Un algoritmo sencillo para la eliminación de puntos outliers es considerar para cada partición \widehat{Sp}_i la distancia mínima d_{min}^i de los puntos de dicha partición al segmento asociado. Luego mediante un factor $\delta > 1$ se obtiene la distancia máxima permitida para la partición \widehat{Sp}_i dada por $d_{max}^i = d_{min}^i \cdot \delta$. De esta manera se obtiene para cada \widehat{Sp}_i un nuevo subconjunto de puntos

$$\widehat{Sp}_i^1 = \left\{ p \in \widehat{Sp}_i \mid d_{p, \overline{p_i, p_{i+1}}} \leq d_{max}^i \right\}$$

donde $d_{p, \overline{p_i, p_{i+1}}}$ denota la distancia del punto p al segmento de poligonal $\overline{p_i, p_{i+1}}$.

Finalmente la nueva función de costo basada en esquemas de subdivisión queda determinada por la siguiente ecuación.

$$\mathcal{F}_{PLSF}(Sp, \delta) = \sum_{i=1}^{2^k+1} \sum_{p \in \widehat{Sp}_i^1} d_{p, \overline{p_i, p_{i+1}}}^2.$$

Nótese que el particionamiento puede hacerse durante el proceso de subdivisión lo cual implica un costo computacional de orden lineal. De esta forma cada punto del conjunto Sp es evaluado solo en k rectas para determinar al subconjunto que pertenece, lo cual es muy eficiente computacionalmente.

Este enfoque permite una buena aproximación de la distancia de un punto a la elipse a la vez que es una herramienta favorable para la eliminación de puntos outliers.

En lo adelante al algoritmo basado en la función de costo que considera el uso de esquemas de subdivisión se le denomina “Polinomial Least Square Fitting” (PLSF).

4. ESTIMACIÓN DE TANGENTES EN DATOS DISPERSOS

El objetivo es estimar la tangente en un punto $P^* \in Sp$ mediante el algoritmo de estimación de tangentes en datos discretos propuesto por Matas [14], o por el algoritmo de Albrecht [1].

Con este fin se considera una ventana de trabajo de tamaño W centrada en P^* . Sea Ω el conjunto de puntos de Sp que se encuentran dentro de esta ventana de trabajo,

$$\Omega = \{P \in Sp : |P_x - P_x^*| \leq W \wedge |P_y - P_y^*| \leq W\}. \quad (12)$$

Centrado en P^* , se pueden definir cuatro direcciones indicadas por los vectores

$$\begin{aligned} v_1 &= (0, 1), & v_2 &= (1, 0), \\ v_3 &= (1, 1), & v_4 &= (-1, -1). \end{aligned} \quad (13)$$

Considerando solo un vector director se tiene que la recta que pasa por P^* , perpendicular a la pendiente de dicho vector director, divide al plano en dos semiplanos \mathfrak{R}_1 y \mathfrak{R}_2 y por lo tanto, divide a Sp en dos conjuntos disjuntos,

$$\begin{aligned} Sp_1 &= \{P \in \mathfrak{R}_1^{v_i} \cap \Omega\}, \\ Sp_2 &= \{P \in \mathfrak{R}_2^{v_i} \cap \Omega\}. \end{aligned} \quad (14)$$

Se desea encontrar k puntos en Sp_1 y k puntos en Sp_2 , de forma tal que P^* permanezca en el medio de los $2k + 1$ puntos.

Centremos nuestra atención en el caso de encontrar los k puntos en el conjunto Sp_1 respecto al vector director v_i , el análisis para el conjunto Sp_2 es totalmente análogo usando el vector dirección $-v_i$.

Sea P_m con $0 < m < k$, el punto m -ésimo encontrado por este método. Se define $P_0 = P^*$ como punto inicial, $Sp_1^m = Sp_1$ y $P_n \in Sp_1^m$ el punto que cumple

$$d(P_n, P_m) \leq d(P, P_m), \quad \forall P \in Sp_1^m, \quad (15)$$

con $d(\cdot, \cdot)$ métrica euclídeana. Usando la condición (15) se selecciona P_n como el punto $(m + 1)$ -ésimo a considerar para estimar la tangente en P^* .

Una vez que se tiene el punto $(m + 1)$ -ésimo, se eliminan de Sp_1^m los puntos que quedan al mismo lado que el punto P^* con respecto a la recta perpendicular a $\overline{P_n P^*}$. Este filtrado de los puntos de Sp_1^m evita la introducción de formas de Z al no permitir la búsqueda del próximo punto en dirección opuesta o perpendicular a la dirección de análisis.

La condición de parada de este método es cuando se encuentran k puntos en Sp_1 y k puntos en Sp_2 para alguno de los vectores directores. De no poder encontrar estos puntos, se propone ampliar la ventana de trabajo W y/o disminuir el valor de k .

5. EXPERIMENTOS

Para evaluar la calidad de los métodos propuestos, se comparan los parámetros obtenidos mediante el algoritmo de ajuste con los coeficientes de la elipse experimental a partir de la cual se generan los puntos a ajustar. Esta comparación se realiza mediante el cálculo de la distancia euclídeana entre las componentes de la elipse experimental y la obtenida mediante el proceso de ajuste. Si se obtiene un error en la aproximación menor que un error prefijado, se considera que el ajuste fue exitoso.

Los experimentos están divididos en dos clases:

1. En el primer experimento, se pretende observar el comportamiento de los algoritmos propuestos ante la presencia de diferentes niveles de ruido Gaussiano.
2. En un segundo experimento, se pretende observar cuán robustos son los algoritmos propuestos ante la presencia de puntos atípicos en los datos.

Para cada uno de los experimentos se realizaron 1000 corridas. Las estadísticas que se muestran a continuación representan los porcentos donde el ajuste se consideró exitoso.

En la Tabla 1 se observan los resultados del primer experimento. Se presentan los porcentos de ajustes exitosos que obtiene cada algoritmo para diferentes niveles de ruido Gaussiano.

Algoritmo	Niveles de ruido Gaussiano						
	0	0,5	1,0	1,5	2,0	2,5	3,0
PLSF	99,5	97,6	93,5	87,4	81,3	74,9	70,0
OHF	99,6	98,2	95,3	93,7	89,5	83,1	79,8
DLSF	100	100	100	100	100	100	100

Cuadro 1: Porcentaje de ajustes exitosos de los algoritmos sujeto a un $error = 5$.

Los mejores resultados en ambientes donde existe ruido Gaussiano los obtiene el algoritmo DLSF, mientras que el algoritmo OHF muestra un comportamiento ligeramente superior respecto a los resultados obtenidos por PLSF. Justamente una de las ventajas principales que tiene el algoritmo DLSF es su estabilidad ante presencia de ruido Gaussiano. Los algoritmos propuestos, aunque tienen estadísticas inferiores al DLSF, mantienen un porcentaje de acierto que se puede considerar aceptable dado los niveles de ruido que se han generado en las elipses.

Los resultados del segundo experimento se exponen en la Tabla 2. Se presentan los porcentos de ajustes exitosos que obtienen los algoritmos para diferentes niveles de ruido Gaussiano en presencia del 1% de valores atípicos. La Figura 2 ilustra el comportamiento de los algoritmos.

Algoritmo	Niveles de ruido Gaussiano						
	0,0	0,5	1,0	1,5	2,0	2,5	3,0
PLSF	89,8	86,2	81,8	76,6	70,1	59,9	52,0
OHF	85,7	82,8	80,3	78,7	72,8	67,3	59,5
DLSF	35,5	34,6	34,8	34,3	34,8	33,2	36,0

Cuadro 2: Porcentaje de ajustes exitosos con 1% de valores atípicos, sujeto a un $error = 5$.

Este experimento muestra un gran descenso en los porcentos de los resultados exitosos del algoritmo DLSF, aproximadamente entre el 64% y el 67%, mientras que se obtiene un descenso máximo del 18% para PLSF y del 20% para OHF, mostrando la superioridad de los algoritmos propuestos respecto a DLSF para un bajo porcentaje de valores atípicos. En las aplicaciones la presencia de valores atípicos es una realidad que afecta de manera sustancial a la mayoría de los algoritmos, los métodos basados en mínimos cuadrados tienen un punto de ruptura¹¹ del 0%, o sea, bajo la presencia de valores atípicos con estos métodos se obtienen malos resultados, por lo que es necesario para su aplicación en estos casos filtrar los datos de antemano, lo cual aumenta sustancialmente el costo computacional requerido para dar una aproximación aceptable. Justamente la ventaja de los algoritmos propuestos se basa en que los mismos tienen la capacidad de filtrar los datos a la vez que se realiza el ajuste.

En la Tabla 3 se muestran los resultados referentes a un experimento donde se varía el porcentaje de valores atípicos desde el 2% hasta el 5%.

¹¹Se denomina punto de ruptura al porcentaje máximo de valores atípicos con el cual un algoritmo obtiene buenos resultados.

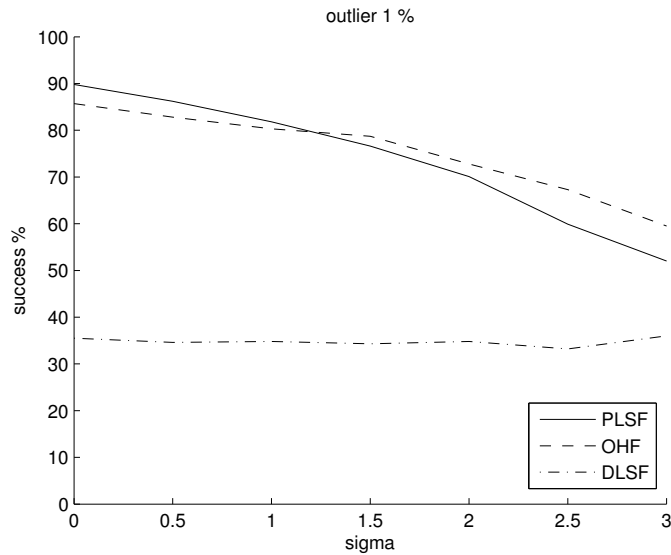


Figura 2: Gráfica comparativa de los algoritmos con 1 % de valores atípicos.

% Valores atípicos	Algoritmo	Niveles de ruido Gaussiano						
		0,0	0,5	1,0	1,5	2,0	2,5	3,0
2	PLSF	98,3	96,3	93,2	87,7	82,1	77,3	70,5
	OHF	95,8	94,0	92,5	88,0	85,8	84,4	79,1
	DLSF	25,0	26,2	27,8	25,9	23,9	26,4	27,1
3	PLSF	95,0	91,5	89,5	86,7	78,7	72,9	68,0
	OHF	91,0	87,9	87,5	86,0	82,6	78,2	73,5
	DLSF	16,6	14,4	14,8	14,1	14,4	15,1	13,8
4	PLSF	87,0	85,5	82,5	77,6	73,6	64,0	63,2
	OHF	83,5	81,4	80,8	76,7	76,2	69,5	69,4
	DLSF	8,7	8,6	8,4	9,5	7,8	9,1	8,5
5	PLSF	80,5	80,1	73,8	70,5	66,5	61,2	56,6
	OHF	76,4	75,4	74,9	68,9	68,9	63,2	61,8
	DLSF	3,4	3,6	4,2	4,7	4,4	4,4	4,7

Cuadro 3: Porcentaje de ajustes exitosos de los algoritmos sujeto a un $error_{ED} = 10$, con valores atípicos desde el 2% hasta el 5%.

El algoritmo DLSF se deteriora de manera sustancial obteniendo como valor máximo de convergencia un 25% y un valor mínimo igual al 3%. El algoritmo PLSF tiene como valor máximo un 98,3% bajo la presencia de 2% de puntos atípicos, para el caso de 5% de puntos atípicos se tiene un 80,5% de casos favorables, y un 56,6% como la peor de las estadísticas registradas en los experimentos. Estos resultados avalan que en situaciones caóticas el algoritmo PLSF tiene mayor robustez que el algoritmo DLSF. Para el caso del algoritmo OHF los resultados son interesantes dado que en comparación con PLSF para iguales niveles de ruido las estadísticas favorecen al enfoque PLSF cuando el

ruido Gaussiano es pequeño, sin embargo al aumentar el nivel de ruido Gaussiano el algoritmo OHF mantiene mejores estadísticas superando al PLSF.

En la Figura 3 se muestran los gráficos correspondientes a los conteos de frecuencia realizados mediante las corridas de cada uno de los tres algoritmos. En correspondencia con la Tabla 3 y la Figura 3, podemos asegurar que los algoritmos PLSF y OHF mantienen un comportamiento adecuado ante la presencia de patrones atípicos.

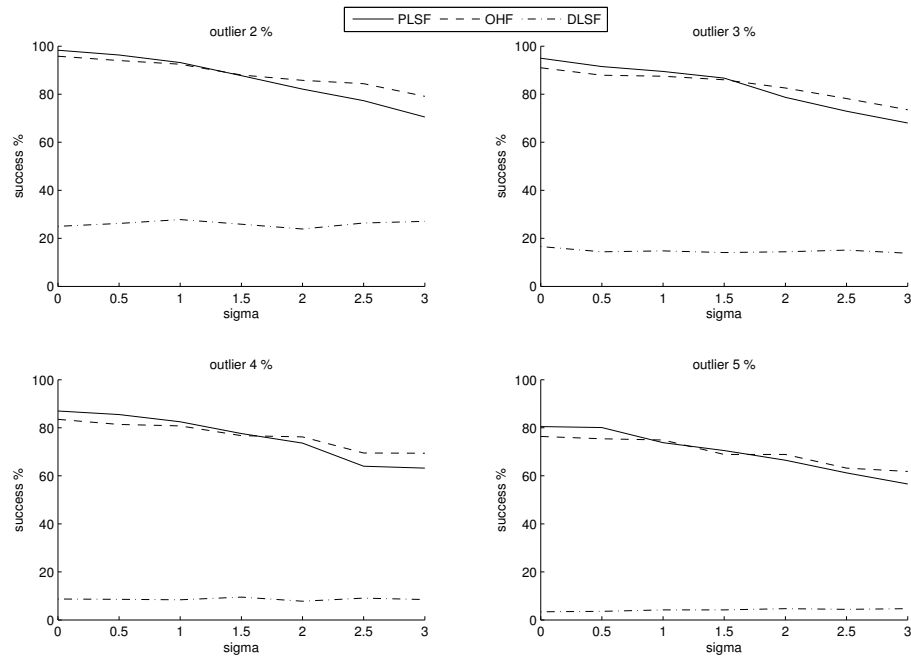


Figura 3: Gráfica comparativa de los algoritmos *PLSF*, *OHF* y *DLSF* con valores atípicos desde el 2 % hasta el 5 %.

La Figura 4 muestra los ajustes realizados por los tres algoritmos que se comparan. Puede observarse que el método DLSF realiza un mal ajuste de la elipse mientras que PLSF y OHF mantienen un comportamiento robusto cuando los valores atípicos aumentan.

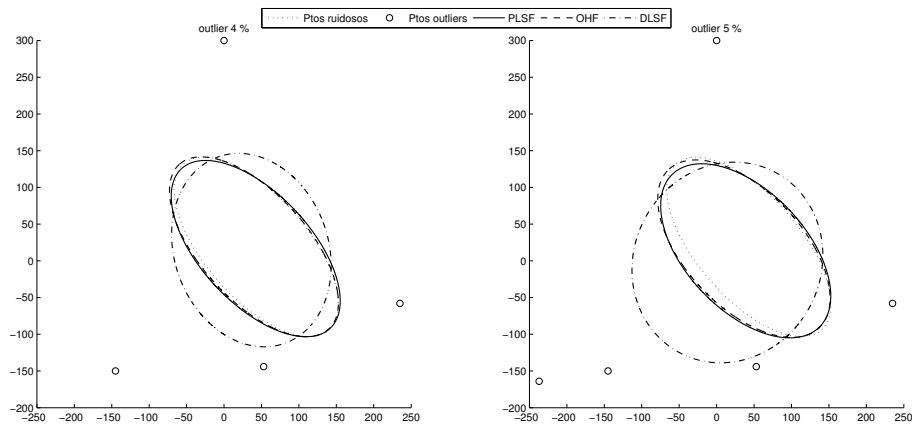


Figura 4: Casos puntuales de ajuste, valores atípicos al 4% y al 5% .

6. CONCLUSIONES

En este documento se han reportado resultados preliminares sobre nuevas estrategias para el ajuste y detección de elipses en datos dispersos. Los métodos propuestos son iterativos pero eficientes y tienen buen comportamiento en presencia de valores atípicos.

La idea fundamental detrás de los nuevos algoritmos es la estimación de tangentes en pares de puntos. La sección cónica que pasa por esos puntos con esas tangentes se puede estimar mediante el método propuesto en este trabajo. Este algoritmo devuelve los seis coeficientes de la forma cuadrática de la cónica. Una de las funciones de costo diseñadas utiliza esta información, la otra no, ya que está basada en el uso de un algoritmo de subdivisión interpolatorio. Los resultados muestran que esta estrategia es eficiente y robusta por lo que merece un estudio posterior. Las funciones de costo diseñadas son univariadas lo que simplifica el problema de optimización.

Se muestra experimentalmente que, en presencia de ruido y valores atípicos en los datos, los algoritmos propuestos superan al algoritmo DLSF. Este último solo supera a los otros en presencia de ruido Gaussiano sin puntos outliers.

RECEIVED: JANUARY, 2013.

REVISED: MARCH, 2014.

Referencias

- [1] ALBRECHT, G., BÉCAR, J. P., FARIN, G., and HANSFORD, D. (2008): On the approximation order of tangent estimators **Computer Aided Geometric Design**, 25:80–95.
- [2] BOOKSTEIN, F. L. (1979): Fitting conic sections to scattered data **Computer Graphics and Image Processing**, 9:56–71.
- [3] DAVE, R. N. and BHASWAN, K. (1992): Adaptive fuzzy c-shells clustering and detection of ellipses **IEEE Trans. Neural Networks**, 3:643–662.
- [4] de la FRAGA, L. G., SILVA, I. V., and CRUZ-CORTEZ, N. (2007): Euclidean distance fit of ellipses with a genetic algorithm **Proceedings of the EvoWorkshops**, 4448:359–366.
- [5] DÍAZ FUENTES, R. (2010): Esquema de subdivisión interpolatorio con parámetros de tensión local basado en spline cónico Master's thesis, Universidad de la Habana Tesis de Licenciatura.

- [6] ESTRADA, S. J. and DÍAZ FUENTES, R. (2010): Esquema de subdivisión interpolatorio con parámetros de tensión local basado en spline cónico Technical report, Instituto de Cibernética, Matemática y Física.
- [7] FARIM, G., HOSCHEK, J., and KIM, M. S. (2002): **Handbook of Computer Aided Geometric Design** Elsevier.
- [8] FITZGIBBON, A. W., PILU, M., and FISHER, R. B. (1996): Direct Least Squares Fitting of Ellipses In **International Conference on Pattern Recognition**.
- [9] FITZGIBBON, A. W., PILU, M., and FISHER, R. B. (1999): Direct Least Squares Fitting of Ellipses **IEEE Trans. Pattern Anal. Mach. Intell.**, 21:476–480.
- [10] GANDER, W., GOLUB, G. H., and STREBEL, R. (1994): Least-Squares Fitting of Circles and Ellipses **BIT**, 43:558 – 578.
- [11] GATH, I. and HOORY, D. (1995): Fuzzy clustering of elliptic ring-shaped clusters **Pattern Recognition Letters**, 16:727–741.
- [12] HALIRr, R. and FLUSSER, J. (1998): Numerically Stable Direct Least-Squares Fitting of Ellipses In **Sixth International Conference. Computer Graphics and Visualization**.
- [13] KHARMA, N., MOGHNIEH, H., YAO, J., GUO, Y., ABU-BAKER, A., LAGANIERE, J., ROULEAU, G., and CHERIET, M. (2007): Automatic segmentation of cells from microscopic imagery using ellipse detection **Image Processing, IET**, 1:39–47.
- [14] MATAS, J., SHAO, Z., and KITTLER, J. (1995): Estimation of Curvature and Tangent Direction by Median Filtered Differencing In **8th Conf. on Image Analysis and Processing**, San Remo.
- [15] OCHOA, A., SOTO, M., and TEJERA, M. (2010): A fitness function model for detecting ellipses with EDAs In **Evolutionary Computation (CEC), 2010 IEEE Congress on**.
- [16] OLeary, P. and Zsombor-MURRAY, P. (2004): Direct and specific least-square fitting of hyperbol and ellipses **Journal of Electronic Imaging**, 13:492–503.
- [17] PORRILL, J. (1990): Fitting ellipses and predicting confidence envelopes using a bias corrected kalman filter **Image Vision and Computing**, 8(1):11401153.
- [18] QIAO, Y. and ONG, S. (2007): Arc-based evaluation and detection of ellipses **Pattern Recognition**, 40:1990–2003.
- [19] ROSIN, P. L. (1993): Ellipse fitting by accumulating five point fits **Pattern Recognition Letters**, 14:661–668.
- [20] ROSIN, P. L. (1998): Ellipse Fitting using Orthogonal Hyperbol and Stirlings Oval **Graphical Models and Image Processing**, 60:209–213.
- [21] ROSIN, P. L. (2007): Evaluating Harker and OLearys distance approximation for ellipse fitting **Pattern Recognition Letters**, 28:1804–1807.
- [22] ROSIN, P. L. and WEST, G. A. W. (1995): Non-parametric segmentation of curves into various representations **IEEE Trans. PAMI**, 17:1140–1153.
- [23] TAUBIN, G. (1991): Estimation of planar curves, surfaces and non-planar space curves defined by implicit equations, with applications to edge and range image segmentation **IEEE PAMI**, 13:1115–1138.
- [24] WERMAN, M. and GEYZEL, G. (1995): Fitting a second degree curve in the presence of error **IEEE Trans. PAMI**, 17(2):207–211.
- [25] WU, W. Y. and WANG, M. J. J. (1993): Elliptical object detection by using its geometric properties **Pattern Recognition**, 26:1499–1509.
- [26] YIN, R. K. K., TAM, P. K. S., and LEUNG, N. K. (1992): Modification of Hough transform for circles and ellipses detection using 2-D array **Pattern Recognition**, 25:1007–1022.
- [27] YUEN, H. K., ILLINGWORTH, J., and KITTLER, J. (1989): Detecting partially occluded ellipses using the Hough transform **Image Vision and Computing**, 7(1):31–37.