

## Clasificación de huecos en superficies de modelos tridimensionales de forma libre utilizando aprendizaje de máquina supervisado

Supervised machine learning for holes classification of three-dimensional free-form models

Pedro Atencio Ortiz\*

*Instituto Técnico Metropolitano, Sede Medellín (Colombia)*

John Branch Bedoya\*\*

*Universidad Nacional de Colombia, Sede Medellín (Colombia)*

Germán Sánchez Torres\*\*\*

*Universidad del Magdalena, Santa Marta (Colombia)*

\* Profesor, Instituto Técnico Metropolitano, Facultad de Ingenierías, Medellín (Colombia). Magíster en Ingeniería de Sistemas. [pedroatencio@itm.edu.co](mailto:pedroatencio@itm.edu.co)

\*\* Profesor asociado, Universidad Nacional de Colombia, Facultad de Minas, Medellín (Colombia). Doctor en Ingeniería de Sistemas. [jwbranch@unal.edu.co](mailto:jwbranch@unal.edu.co)

\*\*\* Profesor asistente, Universidad del Magdalena, Facultad de Ingenierías, Santa Marta (Colombia). Doctor en Ingeniería de Sistemas. [gsanchez@unimagdalena.edu.co](mailto:gsanchez@unimagdalena.edu.co)

**Correspondencia:** Pedro Atencio Ortiz, Calle 54a Sur No. 30-01, Instituto Tecnológico Metropolitano, Medellín - Antioquia. Bloque L, Ofic. 334, Tel: (57-4) 4600727 Ext: 5586. [pedroatencio@itm.edu.co](mailto:pedroatencio@itm.edu.co).

## Resumen

La tarea del llenado de huecos en el proceso de reconstrucción tridimensional requiere de un usuario experto que se encargue de seleccionar los que deben ser corregidos (llenados), en casos en que existan huecos reales en la superficie del objeto que se esté reconstruyendo. Generalmente, los trabajos propuestos en esta área asumen que la superficie de los objetos es continua, por lo que todos los huecos deben ser corregidos. Lo anterior no es cierto para muchos casos, por ejemplo, en piezas industriales y objetos de forma libre. En este trabajo se propone un método para la clasificación automática de huecos en superficies tridimensionales de objetos de forma libre, en dos categorías: real o huecos que no deben ser corregidos y anomalía o huecos que deben ser corregidos. Para ello, se calculan tres características del contorno del hueco: torsión, curvatura y tamaño, y que posteriormente se utilizan en un sistema de clasificación supervisada. Los resultados muestran un porcentaje de clasificación superior al 90%.

**Palabras clave:** aprendizaje supervisado de máquina, llenado de huecos, reconstrucción tridimensional.

## Abstract

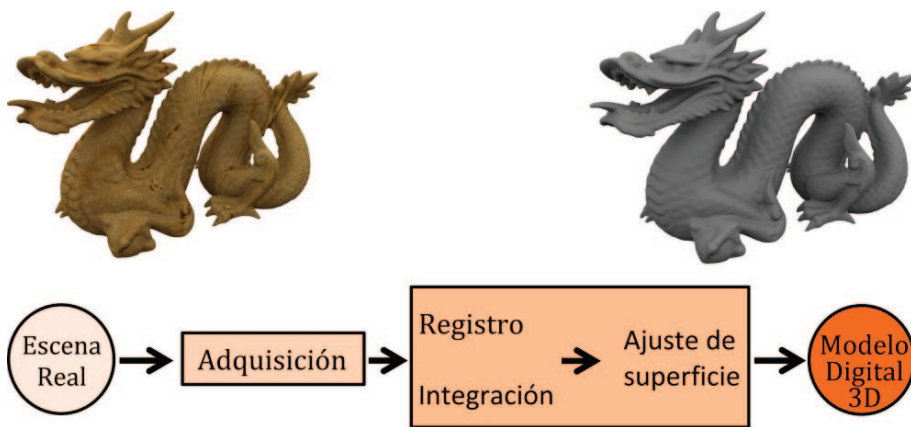
Hole-filling task in tridimensional reconstruction process requires an expert user to select holes to be corrected (filled) in cases where there are real holes in the surface of the object that is being reconstructed. Generally, proposed works in hole-filling tasks assume the surface of the object is continuous, so that all holes must be corrected. The latter is not true for many cases e.i. industrial parts and free-form objects. In this work, it is proposed a method for automatic hole classification in tridimensional surfaces of free-form objects into two categories: real or holes that must not be corrected, and anomalies or holes that must be corrected. For this purpose, three characteristics of hole contour are calculated: torsion, curvature and size, and subsequently are used in a supervised classification system. Results show a classification rate over 90%.

**Keywords:** hole-filling, supervised machine learning, three-dimensional Reconstruction.

*Fecha de recepción:* 02 de julio de 2013  
*Fecha de aceptación:* 15 de diciembre de 2014

## INTRODUCCIÓN

La reconstrucción tridimensional de superficies (RTS) puede ser definida como la digitalización de información de volumen, forma y textura de objetos reales. Es un proceso comúnmente utilizado en distintas áreas del conocimiento y en múltiples aplicaciones. Según Blais [1], la RTS ha superado su etapa de laboratorio de la década los ochenta y de numerosas demostraciones durante los noventa. Sin embargo, esta tecnología está en su periodo de infancia respecto a su contraparte bidimensional. De acuerdo con Besl [2], el proceso de RTS puede ser descompuesto en cuatro etapas principales: *adquisición*, *registro*, *integración* y *ajuste de superficie* (figura 1). La etapa de adquisición captura datos de la superficie del objeto mediante algún sistema de adquisición, del cual se obtienen un conjunto de vistas parciales.



**Figura 1.** Etapas del proceso de reconstrucción tridimensional de superficies.

Exceptuando algunos casos, estas vistas parciales se digitalizan sin un marco de referencia común. Alinear dichas vistas y darles un único marco de referencia es el objetivo de la etapa de registro. La naturaleza parcial del proceso de adquisición y el procedimiento de alineación de datos genera un conjunto de anomalías en el modelo conocidas como *huecos*. Estas anomalías deben ser corregidas en la etapa de integración para generar un conjunto de puntos que será finalmente utilizado en la etapa de ajuste para estimar un modelo computacional, como una malla triangular o una superficie NURBS.

La principal limitación del proceso de RTS es la necesidad de la intervención del usuario en algunas de las etapas intermedias como el registro y la integración. Es por tanto deseable la proposición de una metodología que permita la automatización del proceso. La automatización de la RTS ha sido una tarea parcialmente exitosa en ciertas aplicaciones, pero está muy lejos de presentarse como una solución general, debido principalmente a la complejidad de tratar con las casi ilimitadas configuraciones que puede tomar un objeto de forma libre [2], [3].

En el caso particular de la etapa de integración, en la cual se corrigen las anomalías como los huecos presentes en los datos, la intervención del usuario está orientada a la discriminación manual de las anomalías que se deben corregir y aquéllas que deben conservarse por estar presentes en el modelo real.

Las causas que generan huecos en las superficies digitalizadas están relacionadas tanto con la naturaleza misma del proceso de adquisición como con las propiedades de la superficie del objeto real [4]–[6]. Las oclusiones constituyen la fuente principal de generación de huecos durante la etapa de adquisición, y pueden ser entendidas como: “Una obstrucción total o parcial de una vista del objeto, producida por una región perteneciente al mismo, lo cual se presenta en gran parte debido a la presencia irregularidades en la superficie de objetos geoméricamente complejos, las cuales generan hendiduras profundas y/o protuberancias que pueden producir regiones de difícil acceso para el sensor en el proceso de triangulación” [7] (figura 2).

La comunidad científica del área ha enfocado gran parte de sus esfuerzos en la corrección de huecos en la etapa de integración, debido a que constituyen el conjunto de anomalías más frecuentes en la RTS [3], [4]. Los métodos propuestos para la corrección o llenado de huecos (*hole-filling*) se pueden agrupar en dos categorías principales: métodos de corrección global [8]–[12] y métodos de corrección local [5], [13]–[17].

El enfoque global consiste en estimar una función de interpolación o aproximación utilizando todo el conjunto de puntos adquiridos para generar un nuevo conjunto de puntos que no contenga huecos. Estos trabajos consideran que el objeto final reconstruido es completamente cerrado, pero esta se convierte en su mayor limitación al no permitir reconstruir con fidelidad

objetos que presenten discontinuidades. Por el contrario, en el esquema local, una vez se detectan los huecos en la superficie, éstos son corregidos tomando como referencia sólo la región circundante al contorno del hueco. Los datos del contorno son utilizados para construir funciones de aproximación de la región faltante y generar datos al interior del hueco para posteriormente integrarlos al modelo digital. Sin embargo, tales métodos requieren intervención manual en la selección de los huecos a corregir, lo que impide automatizar el proceso de RTS.

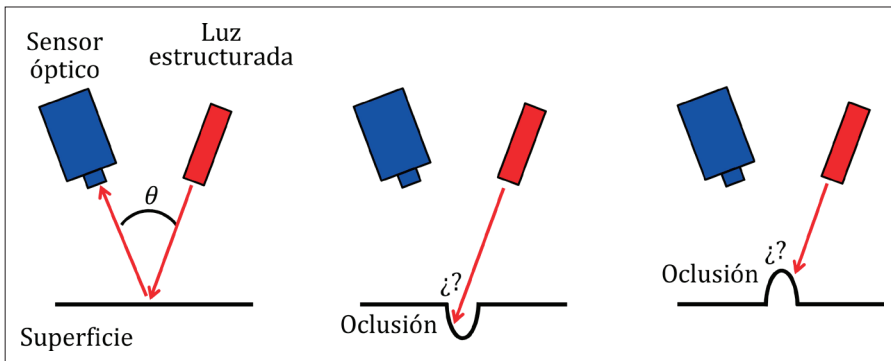


Figura 2. Causas de oclusiones en RTS.

Una solución a esta problemática es proponer un método de discriminación automática que remplace la necesidad de la intervención manual. La discriminación de huecos ha sido abordada mediante el estudio de diferentes características geométricas, como la curvatura y la torsión estimadas en el borde del hueco [18], [19]. Sin embargo, estos trabajos están limitados a la discriminación mediante el uso de un umbral experimental que requiere calcularse de acuerdo con el dominio geométrico específico de los objetos tratados.

En este trabajo se propone un método dirigido a la discriminación automática de la naturaleza de los huecos detectados en un conjunto de puntos tridimensionales que representan un objeto. El método propuesto discrimina los huecos analizando las características geométricas del contorno mediante el uso de algoritmos de aprendizaje supervisado de máquina. El análisis de las propiedades geométricas permite a los algoritmos de aprendizaje detectar diferencias en las variaciones de las magnitudes y sus comportamientos. El

trabajo considera que los huecos que constituyen una anomalía presentarán una mayor variación en sus características geométricas que aquéllos que están presentes en el objeto real.

## METODOLOGÍA

El método propuesto inicia con la identificación del hueco en la superficie tridimensional y la posterior extracción de los vértices del contorno, los cuales son evaluados en relación con sus tamaños y un valor umbral  $t$  definido por el usuario. Si el tamaño del contorno es menor al valor  $t$ , entonces será clasificado como *anomalía*. En caso contrario, los vértices del contorno son escalados y este es caracterizado estimando la varianza de la torsión y la varianza de la curvatura. El proceso de clasificación se realiza mediante una máquina de vectores de soporte con kernel lineal que determina si el contorno pertenece a la clase *real* o a la clase *anomalía* (figura 3).

La aproximación descriptiva de la clasificación en el reconocimiento de patrones radica en obtener un vector de características relevantes que describan un objeto y asignar una categoría a dicho objeto, basado en un conocimiento previo. Sin embargo, encontrar tal conjunto de características es un problema abierto en la literatura, pues depende en gran parte del conocimiento que el experto humano tenga del objeto de estudio. En este sentido, el vector característico del contorno del hueco está compuesto por la torsión, la curvatura y el tamaño.

### Identificación de huecos

El primer paso consiste en identificar el conjunto de aristas del contorno de cada hueco. En una superficie triangular, un hueco puede ser definido como un ciclo cerrado de aristas y vértices límite. Una arista límite es aquella perteneciente sólo a un triángulo [20]. La aproximación más simple consiste recorrer el conjunto de puntos o vértices de la superficie hasta identificar un vértice límite  $y$ , partiendo de este, se procede a encontrar el conjunto de aristas límites conectadas a dicho vértice. Si el conjunto de aristas límite genera un ciclo cerrado, entonces éstas conforman el borde de un hueco en la superficie. El proceso anterior se repite para todos los vértices límite de la superficie.

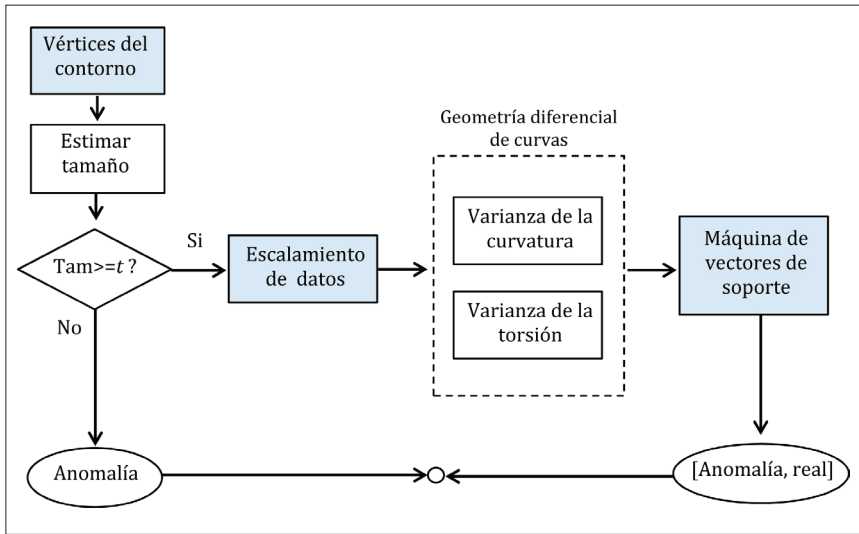


Figura 3. Resumen del método propuesto.

### Escalado de los vértices del contorno

Debido a que el valor de la curvatura es dependiente del valor (orden) de los vértices del contorno y que la estimación de la torsión depende de la curvatura, es necesario escalar el volumen del contorno al valor 1. Para ello se calcula una constante  $k$  (ecuación 1) que al ser multiplicada por el contorno  $V$  resulta en un nuevo contorno  $VSc$  cuyo volumen es 1 (ecuación 2).

$$k = \sqrt[3]{\frac{1}{(maxx - minx) * (maxy - miny) * (maxz - minz)}} \quad (1)$$

$$VSc = V * k \quad (2)$$

### Estimación del tamaño

El tamaño del hueco es una propiedad comúnmente utilizada para determinar si éste debe ser corregido o conservado. Frecuentemente, el tamaño del hueco es equivalente al número de aristas que lo conforman; sin embargo, determinar si un hueco es pequeño o no a partir de esta medida requiere tener en cuenta la resolución de la superficie en que éste se encuentra. Para

ello, un usuario experto ingresa un valor de umbral de tamaño  $t$ , el cual es comparado con el número de aristas  $N_a(t)$  de cada hueco  $h_i$  para decidir si se debe corregir o no (ecuación 3). Por lo tanto, la medida de tamaño es definida como el cociente del número de aristas del contorno de hueco entre el número total de aristas del objeto (ecuación 4).

$$N_a(h_i) \leq t \begin{cases} \text{verdadero} & \rightarrow \text{corregir} \\ \text{falso} & \rightarrow \text{conservar} \end{cases} \quad (3)$$

$$S(h_i) = \frac{N_a(h_i)}{N_a(\text{surf})} \quad (4)$$

Donde  $S(h_i)$  es el tamaño del contorno  $h_i$  de un hueco en una superficie  $\text{surf}$  y  $N_a(h_i)$  y  $N_a(\text{surf})$  corresponden al número de aristas del contorno  $h_i$  y la superficie  $\text{surf}$ , respectivamente.

#### Estimación de la torsión y la curvatura

Según la geometría diferencial de curvas, una curva puede ser descrita a través de su curvatura y su torsión. La primera describe el cambio del vector tangente con respecto a la longitud de la curva, la segunda mide alguna magnitud que indica la rotación del plano osculador, que está conformado por los vectores normal y tangente con respecto a la longitud de la curva (figura 4).

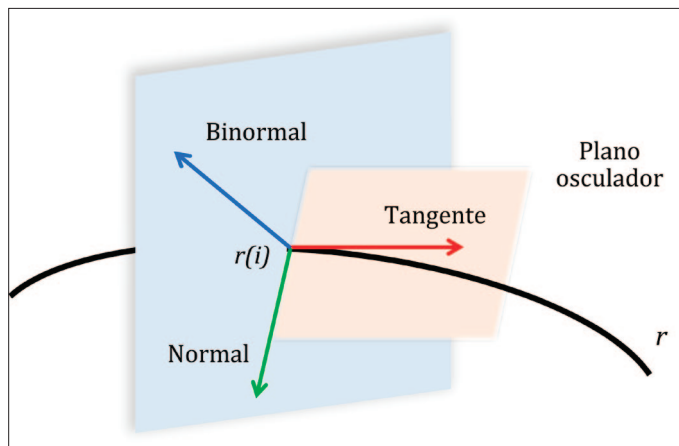


Figura 4. Plano osculador de la curva paramétrica  $r$  en el punto  $r(i)$ .



Ambas propiedades han sido utilizadas en algunos trabajos recientes con el fin de caracterizar la distorsión de los contornos de huecos en superficies de objetos de forma libre [18], [19]. Debido a que los contornos de los huecos son discretos, la aplicación de la geometría diferencial de curvas en dichos casos se realiza mediante el ajuste de una curva paramétrica al conjunto de puntos del contorno del hueco y posteriormente se estima la torsión y la curvatura sobre la curva aproximada. Esta aproximación presenta ciertas limitaciones, por ejemplo, aproximar una curva paramétrica implica la selección del tipo y orden de la curva de acuerdo al problema. Segundo, el tiempo computacional de esta aproximación generalmente es costoso, más cuando se trata de un modelo tridimensional de alta resolución. Por último, ajustar una curva paramétrica tiende a generar un suavizado de los puntos originales, lo cual resulta desfavorable a la hora de obtener una medida de distorsión.

Para la estimación de la torsión y curvatura puntuales del contorno del hueco, se utiliza el método propuesto en [21], el cual no requiere realizar un previo ajuste de curvas, con lo que se obtienen mejores resultados en cuanto a precisión y tiempo computacional respecto a los métodos tradicionales. Según An *et al.* [21], la derivada de una curva en un punto específico  $P_k$  puede ser definida como la pendiente de la tangente que pasa por dicho punto, donde la tangente es la línea recta que pasa por  $P_k$  y que minimiza la suma de los cuadrados de las distancias sobre el eje  $y$  entre ésta y los puntos vecinos  $\{p_j = (x_j, y_j) \mid k - m \leq j \leq k + m\}$ .

**Definición 1.** Sea  $y_i = fd(x_i)$  una función discreta. La derivada discreta de dicha función  $y'_k = fd'(x_k)$  viene dada por:

$$y'_k = fd'(x_k) = \frac{\sum_{j=k-m}^{k+m} (x_j - x_k)(y_j - y_k)}{\sum_{j=k-m}^{k+m} (x_j - x_k)^2} \quad (5)$$

**Definición 2.** Sea  $t_i$  el parámetro acumulativo del largo del segmento en el punto  $p_i$  definido como:

$$t_i = \begin{cases} \frac{\sum_{j=1}^{i-1} |p_{j+1} - p_j|}{\sum_{j=1}^{n-1} |p_{j+1} - p_j|} (2 \leq i \leq n) \\ t_1 = 0 \end{cases} \quad (6)$$

**Definición 3.** Sea  $\alpha_d(t_i)$  una curva discreta parametrizada mediante el largo del segmento:

$$t_d(t_i) = \frac{\alpha'_d(t_i)}{|\alpha'_d(t_i)|} \quad (7), \text{ es el vector tangente unitario de la curva discreta } \alpha_d(t_i).$$

$$k_d(t_i) = \frac{|t'_d(t_i)|}{|\alpha'_d(t_i)|} \quad (8), \text{ es la curvatura de la curva discreta } \alpha_d(t_i).$$

$$\beta_d(t_i) = \frac{t'_d(t_i)}{k_d(t_i)|\alpha'_d(t_i)|} \quad (9), \text{ es el vector normal principal de la curva discreta } \alpha_d(t_i).$$

$$b_d(t_i) = t'_d(t_i) \times \beta_d(t_i) \quad (10), \text{ es el vector binormal de la curva discreta } \alpha_d(t_i).$$

$$\tau_d(t_i) = \frac{b'_d(t_i) \cdot \beta_d(t_i)}{|\alpha'_d(t_i)|} \quad (11), \text{ es la torsión de la curva discreta } \alpha_d(t_i).$$

En la definición 3, la derivada de la curva paramétrica  $\alpha'_d(t_i)$  puede ser calculada mediante la definición 1. Por ejemplo,  $\alpha'_d(t_i) = (x'_d(t_i), y'_d(t_i), z'_d(t_i))$ , donde:

$$x'_d(t_i) = \frac{\sum_{j=i-m}^{i+m} (t_j - t_i)(x_j - x_i)}{\sum_{j=i-m}^{i+m} (t_j - t_i)^2} \quad (12)$$

$$y'_d(t_i) = \frac{\sum_{j=i-m}^{i+m} (t_j - t_i)(y_j - y_i)}{\sum_{j=i-m}^{i+m} (t_j - t_i)^2} \quad (13)$$

$$z'_d(t_i) = \frac{\sum_{j=i-m}^{i+m} (t_j - t_i)(z_j - z_i)}{\sum_{j=i-m}^{i+m} (t_j - t_i)^2} \quad (14)$$

Los objetos de forma libre presentan superficies suaves y diferenciables [2], por lo tanto, el contorno de un hueco (real) en un objeto de este tipo debería tener cambios suaves en su curvatura y torsión. Para medir el cambio de la

torsión y curvatura del contorno del hueco, se propone la varianza de los valores de curvatura y de torsión.

$$C(\alpha_d) = \sigma(k_d(\alpha_d)) \quad (15)$$

$$T(\alpha_d) = \sigma(\tau_d(\alpha_d)) \quad (16)$$

Donde  $C(\alpha_d)$  y  $T(\alpha_d)$  corresponden a los descriptores relacionados con la curvatura y la torsión de la curva  $\alpha_d$ , respectivamente.

**Tabla 1.** Descriptores (características) utilizadas para la clasificación del contorno del hueco

Característica	Descripción
Tamaño	Número de aristas del contorno del hueco entre el total de aristas de modelo tridimensional.
Varianza torsión	Varianza del vector de torsión obtenido mediante geometría diferencial de curvas. Esta característica indica qué tan regularmente se generan torsiones en el contorno del hueco.
Varianza curvatura	Varianza del vector de curvatura obtenido mediante geometría diferencial de curvas. Esta característica indica qué tan suave cambia la curvatura del contorno del hueco.

Una vez definido el vector característico de cada contorno, la separabilidad de las clases *anomalía* y *real* es realizada utilizando diferentes métodos de clasificación supervisada, como el análisis discriminante lineal, el perceptrón simple, el método de Golub y las máquina de vectores de soporte con kernel lineal.

Para la experimentación se utilizó un conjunto de datos de 63 contornos de huecos en superficies tridimensionales, cada hueco perteneciente a una de dos clases: anomalía (1) con 22 muestras o hueco real (0) con 41 muestras. Este conjunto de datos fue obtenido de un grupo de 7 objetos escaneados mediante un sistema *Minolta vivid 9i* y de las bases de datos de objetos tridimensionales de Stanford [22], [23].

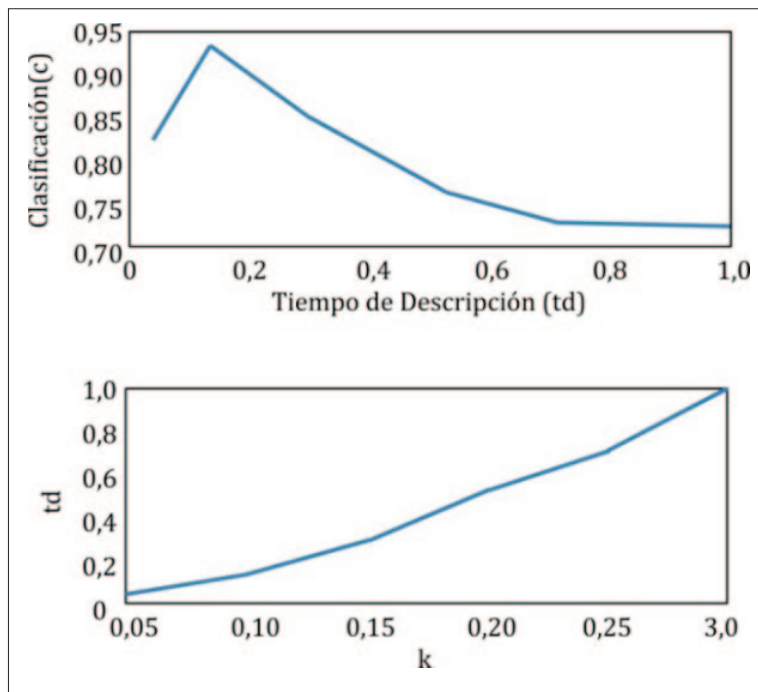
Los algoritmos propuestos de descripción del contorno – *varianza de la curvatura*, *varianza del contorno* y *tamaño* – fueron desarrollados en Matlab; y los clasificadores fueron implementados en el lenguaje Python, utilizando la librería MLPY (*Machine Learning PYTHON*).

## RESULTADOS Y DISCUSIÓN

Para el cálculo de los parámetros acumulativos del largo del segmento y de la derivada discreta, descritos en las ecuaciones (5) y (6), se requiere definir un vecindario  $m$  en número de puntos para cada punto del contorno. El valor de  $m$  fue definido como un porcentaje  $k$  del total de vértices del contorno  $n_{contorno}$  (ecuación 17).

$$m = \text{round}(n_{contorno} * k), \quad k = [0, 1] \quad (17)$$

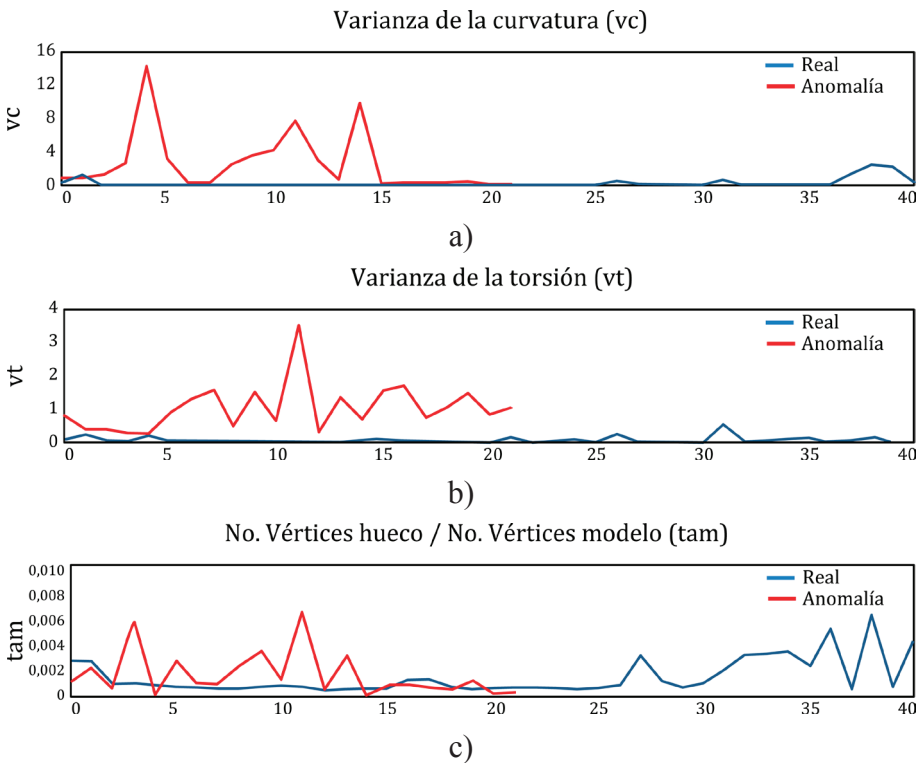
En la figura 5 se puede observar la relación del costo computacional en la estimación de los descriptores contra el nivel de clasificación, al utilizar ambos descriptores en un sistema de clasificación lineal, para  $k=\{0,05, 0,10, 0,15, 0,20, 0,25, 0,30\}$ .



**Figura 5.** Relación precisión de clasificación versus tiempo de descripción (arriba). Relación tiempo de descripción versus parámetro  $k$  (abajo).

El valor de  $k$  se estableció en  $k = 0,10$  de los puntos totales del contorno, debido a que con este valor se encontró la mejor relación *clasificación – tiempo de descripción* para el sistema (figura 5a). Sin embargo, el costo computacional en la estimación de los descriptores de geometría diferencial de curvas crece a medida que aumenta el parámetro  $k$ , como se puede observar en la figura 5b.

En la figura 6 se muestra el comportamiento de los descriptores para cada uno de los 63 contornos del conjunto de datos. Nótese que los descriptores  $vc$  y  $vt$  toman un mayor valor para los contornos de la clase *anomalía* que los pertenecientes a la clase *real*. Sin embargo, el descriptor  $tam$  muestra valores similares para ambas clases. Para validar esto se realizó una selección de características mediante el algoritmo *Iterative Relief* [24], el cual asigna un peso a cada descriptor, proporcional a su relevancia en la separación de las clases. La estimación de estos pesos se puede observar en la tabla 2.



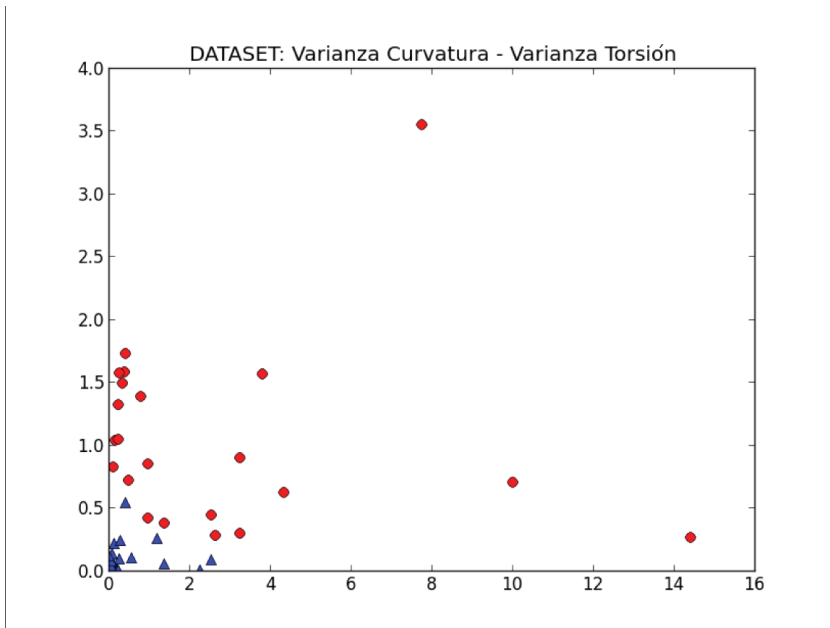
**Figura 6.** Comparación de los descriptores propuestos: a) varianza de la curvatura, b) varianza de la torsión, c) tamaño del contorno.

**Tabla 2.** Pesos asignados a cada descriptor mediante el algoritmo Iterative Relief

Varianza curvatura	Varianza torsión	Tamaño
0,68965431	0,72413875	0,0

De acuerdo con los valores obtenidos, el descriptor *tamaño* no es relevante para la clasificación de los contornos, es decir, los valores que toma dicho descriptor en el conjunto de datos utilizado no brinda información para separar los contornos en sus respectivas clases (figura 6c). Sin embargo, en este trabajo se considera necesario y se propone como un filtro al inicio del proceso de clasificación, con un valor determinado por el usuario según su experiencia acerca de los objetos que está reconstruyendo.

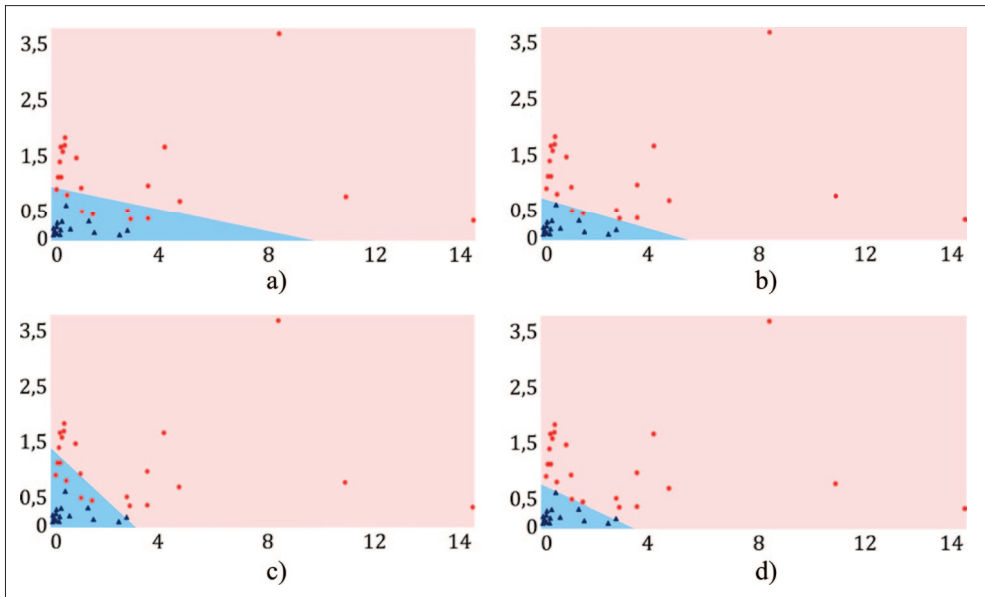
En la figura 7 se muestran los valores de los descriptores de geometría diferencial de curvas para los 63 contornos de trabajo. Se puede observar que la separación de las clases es lineal y que los contornos pertenecientes a la clase real se mueven en un rango menor de valores en ambos descriptores.



**Figura 7.** Valores de varianza de curvatura y varianza de torsión para los contornos de la clase real (triángulo azul) y de la clase anomalía (círculo rojo).

Con el fin de cuantificar la descripción anterior, se utilizaron 3 clasificadores lineales no paramétricos: *análisis discriminante lineal*, Golub [25] y *máquina de vectores de soporte* (kernel lineal), y un clasificador lineal paramétrico *perceptrón simple*. Este último clasificador fue entrenado con los parámetros  $\alpha=0,1$  (tasa de aprendizaje),  $\delta=0,1$  (error mínimo 10%) y  $\beta=1.000$  (número máximo de épocas de entrenamiento).

La precisión de cada clasificador fue calculada utilizando validación cruzada *k-fold* con  $k=1$ , conocida como *leave-one-out*, la cual permite obtener medidas de precisión cuando se cuenta con un número reducido de muestras. En la figura 7 se muestran los resultados de la validación cruzada para cada uno de los clasificadores mencionados; y en la tabla 3 se muestran los valores de precisión obtenidos mediante la técnica *leave-one-out*.



**Figura 8.** Resultado de clasificación para los descriptores de geometría diferencial de curvas: a) análisis discriminante lineal, b) perceptrón simple, c) método de Golub, d) máquina de vectores de soporte con kernel lineal.

La máquina de vectores de soporte con kernel lineal mantiene el mejor resultado de clasificación, seguido del perceptrón simple, ambos con valores superiores al 90%; sin embargo, la precisión del perceptrón simple

disminuyó y la de la máquina de vectores de soporte se mantuvo. Debido a que el problema anterior es de tipo clasificación binaria y a que el número de muestras en ambas clases es distinto, es importante obtener las matrices de confusión con el fin de tener información acerca del sesgo de los clasificadores. Este sesgo es importante porque permite saber cuál es el riesgo de la clasificación. Para este trabajo se decidió seleccionar el clasificador con menor sesgo.

**Tabla 3.** Validación cruzada *leave-one-out* para los clasificadores lineales utilizados

Análisis discriminante lineal	Perceptrón simple	Método de Golub	Máquina de vectores de soporte lineal
0,889	0,937	0,889	0,968

En la tabla 4 se observan las matrices de confusión para cada clasificador, para las cuales se tomó la clase anomalía como casos *positivos* y la clase real como casos *negativos*. Según estas matrices de confusión, los clasificadores con menor sesgo son el perceptrón simple y la máquina de vectores de soporte. Sin embargo, el perceptrón simple tiene un sesgo hacia la detección de la clase *real* (FN=2), mientras que la máquina de vectores de soporte presenta un sesgo igual para ambas clases (FN = FP=1), además de mantener un porcentaje de clasificación alto.

**Tabla 4.** Matrices de confusión para los 4 clasificadores lineales utilizados

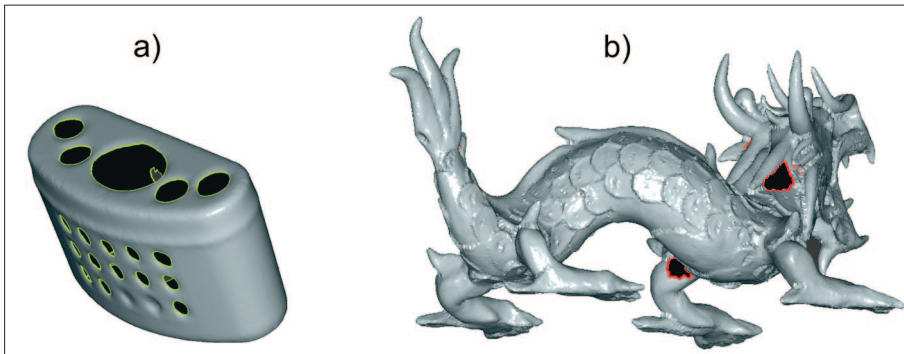
Análisis discriminante lineal			Perceptrón simple		
T (63)	Anomalía (22)	Real (41)	T (63)	Anomalía (22)	Real (41)
Anomalía' (16)	16	0	Anomalía' (20)	20	0
Real' (47)	6	41	Real' (43)	2	41

Golub			Máquina de vectores de soporte lineal		
T (63)	Anomalía (22)	Real (41)	T (63)	Anomalía (22)	Real (41)
Anomalía' (17)	16	1	Anomalía' (22)	21	1
Real' (46)	6	40	Real' (41)	1	40

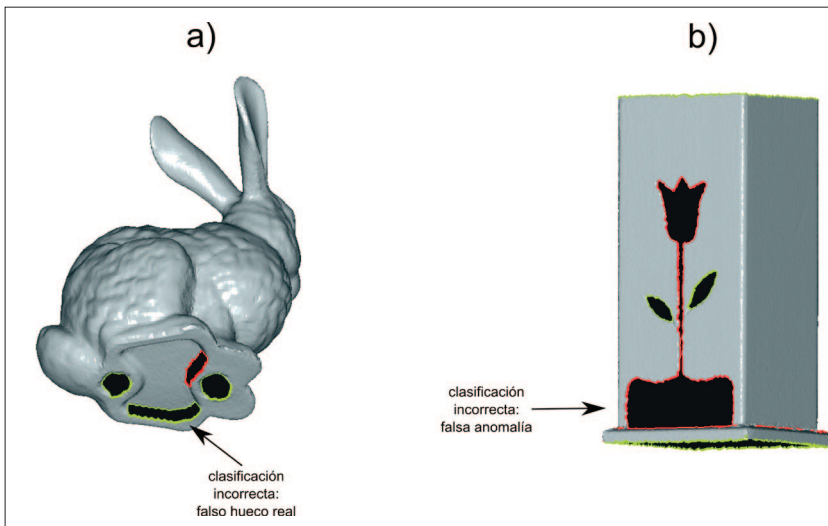
En las figuras 9 y 10 se puede observar el resultado de la clasificación automática de huecos en 2 objetos.





**Figura 9.** Objetos con huecos clasificados de forma correcta.

Obsérvese que el objeto (a) de la figura 9 contiene huecos reales, todos con torsión muy baja, debido a que se encuentran en regiones de la superficie con cambios muy pequeños y en consecuencia son clasificados de forma automática como huecos reales. El objeto (b) de esta misma figura presenta huecos sintéticos, generados mediante un software de procesamiento de mallas triangulares, en regiones de la superficie con cambios abruptos, los cuales son clasificados de forma automática como anomalías. El método propuesto es muy preciso en la detección de anomalías que presenten estas condiciones, comúnmente generados por errores en el proceso de registro de vistas parciales.



**Figura 10.** Objetos con huecos clasificados de forma incorrecta.

En la figura 10 se pueden observar dos objetos que presentan huecos clasificados de forma incorrecta mediante el método propuesto. En el objeto (a) se evidencia una anomalía producto de una oclusión en el proceso de adquisición, la cual ha sido clasificada como un hueco real. Lo anterior ocurre debido a que las anomalías producto de oclusiones en algunas ocasiones son suaves y presentan poca variación en su torsión y curvatura. En el objeto (b) se evidencia un hueco real el cual fue clasificado de forma automática como anomalía. En este caso, dicho hueco presenta una varianza en la torsión muy baja (0,083182) pero una varianza muy alta en su curvatura (2,5338), aunque muy cercana al límite de valor de los huecos reales, posiblemente producto de una adquisición de datos con mucho ruido. Estas clasificaciones incorrectas se pueden observar como el círculo rojo y el triángulo azul que están justo en el borde que separa ambas clases en la figura 8d.

## CONCLUSIONES

En relación con el objetivo de discriminar automáticamente la naturaleza de los huecos en modelos 3D obtenidos en el contexto de la RTS, el descriptor *tamaño* propuesto no mostró ser relevante para la clasificación mediante un sistema de aprendizaje de máquina tipo supervisado.

Por el contrario, los descriptores de geometría diferencial de curvas *varianza de la curvatura* y *varianza de la torsión* permiten separar de forma lineal y con precisión los huecos en las clases *real* y *anomalía*.

Los experimentos muestran resultados de clasificación superiores al 90% y con un sesgo similar para ambas clases, utilizando como clasificador una *máquina de vectores de soporte con kernel lineal*. Aunque el tiempo de ejecución dependerá del sistema de cómputo, resolución del modelo y particularidades de la implementación del método, en la experimentación realizada se obtuvo un tiempo de caracterización y clasificación mínimo de 3,72 ms y 10,4 s para contornos de 17 y 415 vértices respectivamente.

## REFERENCIAS

- [1] F. Blais, "Review of 20 years of range sensor development," *J. Electron. Imaging*, vol. 13, no. 1, 2004. DOI: 10.1117/1.1631921
- [2] P. J. Besl, *The Free-Form Surface Matching Problem In Machine Vision Three Dimensional Scenes*. San Diego: Academic Press, 1990.
- [3] R. J. Campbell and F. J. Patrick, "A Survey of Free-Form Object Representation and Recognition Techniques," *Comput. Vis. Image Underst.*, vol. 81, pp. 166–210, 2001.
- [4] M. Levoy, "Why is 3D Scanning Hard?," in *3D Data Processing, Visualization and Transmission*, 2002.
- [5] S. Marschner, J. Davis, M. Garr, and M. Levoy, "Filling holes in complex surfaces using volumetric diffusion," *SIGGRAPH 2001*, 2001.
- [6] J. Park and A. C. Kak, "3D Modeling of Optically Challenging Objects," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 2, 2008. doi:10.1109/TVCG.2007.1069
- [7] P. Atencio, "Clasificación automática de anomalías relacionadas con ausencia de información en superficies tridimensionales de objetos de forma libre," Universidad Nacional de Colombia, Sede Medellín, 2012.
- [8] H. Hoppe, "Progressive Meshes," *SIGGRAPH '96*, 1996.
- [9] J. Carr, R. K. Beatson, B. C. MacCallum, W. R. Fright, T. J. McLennan, and T. J. Mitchell, "Smooth surface reconstruction from noisy range data," in *GRAPHITE '03 Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, 2003.
- [10] B. S. Morse, T. S. Yoo, P. Rheingans, D. T. Chen, and K. R. Subramanian, "Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions," in *SIGGRAPH '05 ACM SIGGRAPH 2005 Courses*, 2005.
- [11] Y. Ohtake, A. Belyaev, and H.-P. Seidel, "3D scattered data interpolation and approximation with multilevel compactly supported RBFs," *Graph. Model. - Spec. issue SMI 2003*, vol. 67, no. 3, 2005.
- [12] T. Ju, "Robust repair of polygonal models," *ACM Trans. Graph.*, vol. 23, no. 3, 2004. doi:10.1145/1186562.1015815
- [13] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.
- [14] P. Liepa, "Filling holes in meshes," *SGP '03 Proc. 2003 Eurographics/ACM SIGGRAPH Symp. Geom. Process.*, pp. 200 – 205, 2003.

- [15] T. Guo, J. Li, J. Weng, and Y. Zhuang, "Filling Holes in Meshes and Recovering Sharp Edges," *Syst. Man Cybern. 2006. SMC '06. IEEE Int. Conf.*, 2006.
- [16] M. Oliveira and W. Jianning, "A Hole-Filling Strategy for Reconstruction of Smooth Surfaces in Range Images," *XVI Brazilian Symp. Comput. Graph. Image Process.*, 2003.
- [17] A. Kumar and A. Shih, "A Hole-filling Algorithm Using Non-uniform Rational B-splines," in *Proceedings of the 16th International Meshing Roundtable*, 2008.
- [18] J. Branch, P. Boulanger, and F. Prieto, "Automatic hole-filling of triangular meshes using local radial basis function," *3DPVT '06 Proc. Third Int. Symp. 3D Data Process. Vis. Transm.*, pp. 727 – 734, 2006.
- [19] G. Sanchez, J. Branch, and P. Atencio, "A Metric for Automatic Hole Characterization," in *Proceedings of the 19th International Meshing Roundtable*, 2010.
- [20] W. Zhao, S. Gao, and H. Lin, "A robust hole-filling Algorithm for triangular mesh," *Vis. Comput.*, vol. 23, pp. 987–997, 2007.
- [21] Y. An, C. Shao, W. Xiaoliang, and Z. Li, "Geometric properties estimation from discrete curves using discrete derivatives," *Comput. Graph.*, vol. 35, pp. 916–930, 2011.
- [22] Stanford (University), "The Stanford 3D Scanning Repository," 1994. [Online]. Available: <http://www-graphics.stanford.edu/data/3Dscanrep/>.
- [23] Georgia Institute of Technology (Institute), "Large Geometric Models Archive," 2013. [Online]. Available: [http://www.cc.gatech.edu/projects/large\\_models/](http://www.cc.gatech.edu/projects/large_models/). [Accessed: 05-Feb-2013].
- [24] B. Draper, C. Kaito, and J. Bins, "Iterative RELIEF," in *Proceedings of the 2003 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'03)*, 2003, p. 62.
- [25] T. R. Golub, D. K. Slonim, P. Tamayo, and C. Huard, "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Sci. J.*, vol. 286, 1999.