

# Tarea 11 - Simplificación de Geometrías

---

Marco Antonio Esquivel Basaldua

En el área de cartografía la simplificación de geometrías permite representar mapas, ríos, carreteras, entre otros, con una menor resolución a la original pero conservando las formas y la información más representativa del objeto que se esté mostrando. Esto con el fin de, trabajando con imágenes digitales, reducir el tiempo de carga (y descarga) de los datos y el tiempo de renderizado de las imágenes.

Aunque el algoritmo de Douglas-Peucker es el más conocido para la tarea de simplificación de geometrías, el algoritmo de Visvalingam es posiblemente más efectivo y tiene una explicación bastante intuitiva.

En este reporte se explica el algoritmo de Visvalingam y se muestran ejemplos de su implementación simplificando los mapas de los Estados de Aguascalientes, Baja California y Tlaxcala.

## Algoritmo de Visvalingam

---

Propuesto en 1993 por Visvalingam y Whyatt, este algoritmo lleva a cabo simplificación de líneas seleccionando vértices del polígono, o trazo, que serán eliminados, contrario al algoritmo de Douglas-Peucker que selecciona los vértices que serán conservados. El proceso de eliminación se realiza de forma iterativa, en cada iteración se calcula el área del triángulo formado por tres vértices consecutivos, si el esta área es más chica que un área dada de tolerancia el vértice central (de los tres seleccionados para formar el triángulo) es eliminado y una iteración más comienza. Las iteraciones terminan cuando ya no se haya eliminado ningún vértice del polígono o trazo. En la figura 1 se muestra un ejemplo de la implementación de este algoritmo en un trazo para diferentes valores de tolerancia, aumentando ésta de izquierda a derecha.



Figura 1. Ejemplo de la implementación del algoritmo.

Se puede observar en la figura que a medida que se aumenta el valor de la tolerancia se van perdiendo detalles originales del trazo, sin embargo sus características más representativas se conservan.

## Pseudo-código

```

procedimiento Visval (linein: array [1 .. n][2] de reales)
  lineout[1]=linein[1];      El primer punto pasa a la línea de salida
  npin=n;                   Inicialización n° puntos línea de entrada
  npout=1;                   Inicialización n° puntos línea de salida
  mientras (npin>2)
    hallar punto con area efectiva mínima, amin;
    eliminar este punto;
    area[punto]=amin;        Se da un valor de área efectiva a cada punto
  fin de mientras
  desde (i=2;i<=npin-1;i++)
    si (area[i]>=tolerancia)
      npout++;
      lineout[npout]=linein[i];
    fin de si
  fin de desde
  npout++;
  lineout[npout]=linein[npin]; El último punto pasa a la línea de salida
fin del procedimiento

```

## Resultados

El algoritmo de Visvalingam fue aplicado para simplificar las geometrías de los mapas de los Estados de Aguascalientes, Baja California y Tlaxcala, cuya representación está dada por polígonos cuyos vértices son cargados desde un archivo de texto.

Con el objetivo de mostrar una métrica de compresión y similitud se calcula un porcentaje de compresión que muestra el porcentaje de puntos que fueron conservados con respecto al total original y una aproximación al coeficiente de Jaccard que mide la similitud entre el polígono original y el comprimido dividiendo el área del polígono generado sobre la suma del área del mismo y del polígono original, para esto se va a suponer que el polígono generado queda comprendido dentro del polígono original.

Para un área de tolerancia de  $3.4e - 4$  (se utiliza el área de tolerancia como una potencia negativa de  $\pi$  ya que las coordenadas de los puntos están dadas en coordenadas polares) se muestra la figura de la imagen original, la versión simplificada y ambas imágenes empalmadas para una mejor comparación:

## Aguascalientes

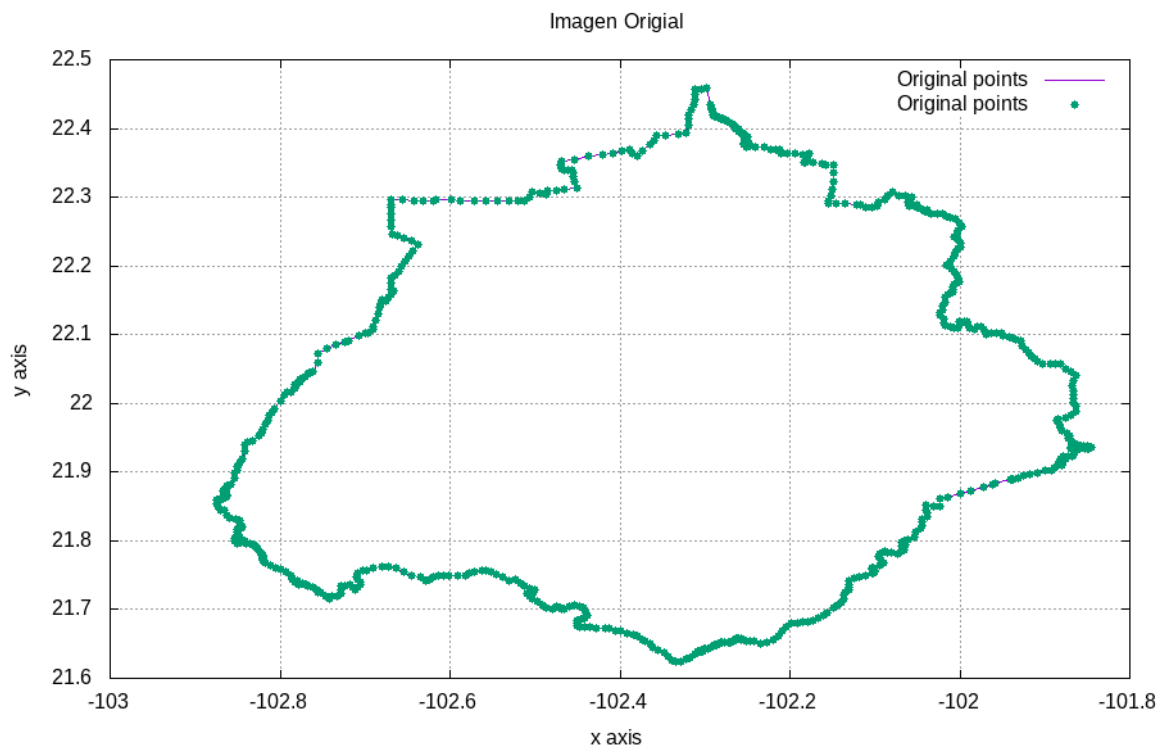


Figura 2. Imagen original con todos los vértices.

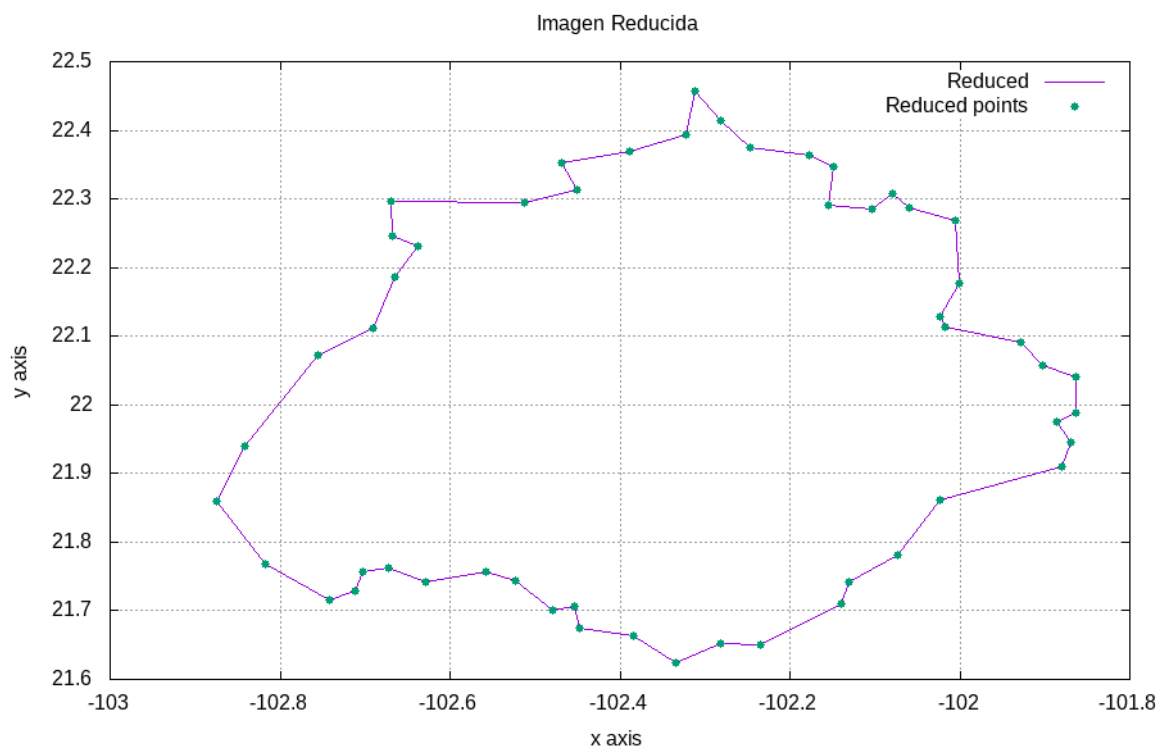


Figura 3. Imagen reducida.

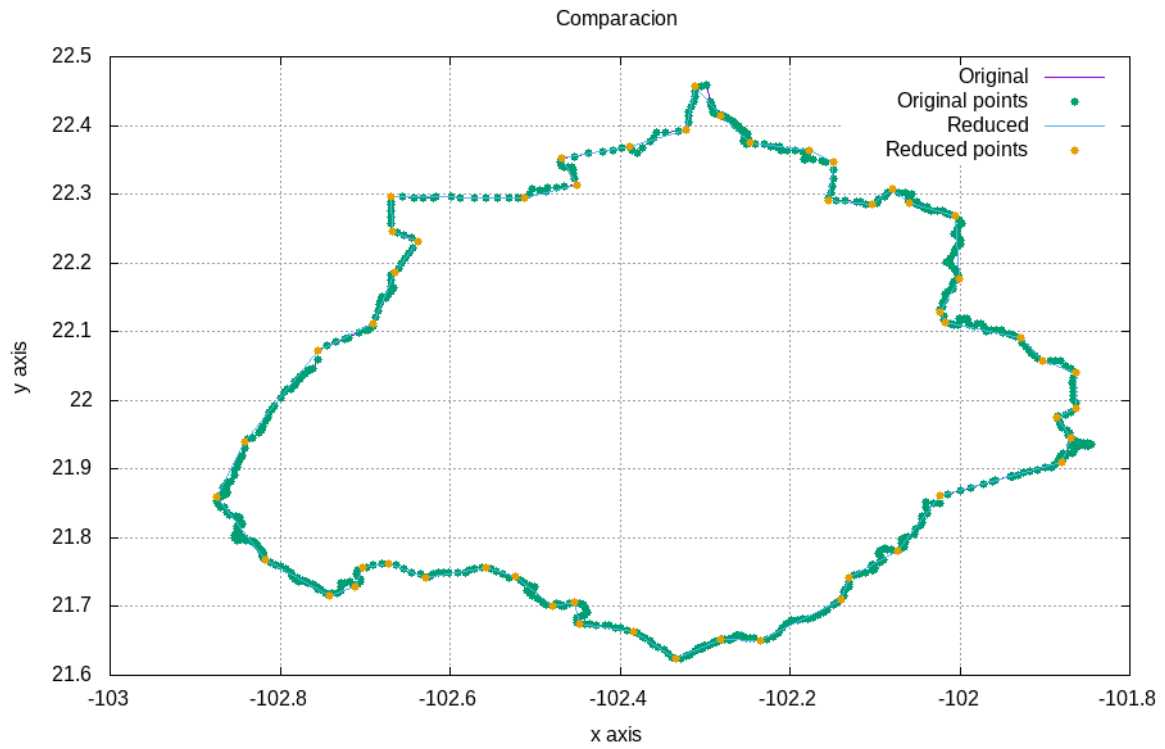


Figura 4. Comparación entre la imagen original y la imagen reducida.

Para este ejemplo las métricas de simplificación son:

- Vértices originales: 660
- Vértices conservados: 52
- Porcentaje de compresión: 7.88 %
- Coeficiente de Jaccart: 1.0

## Baja California

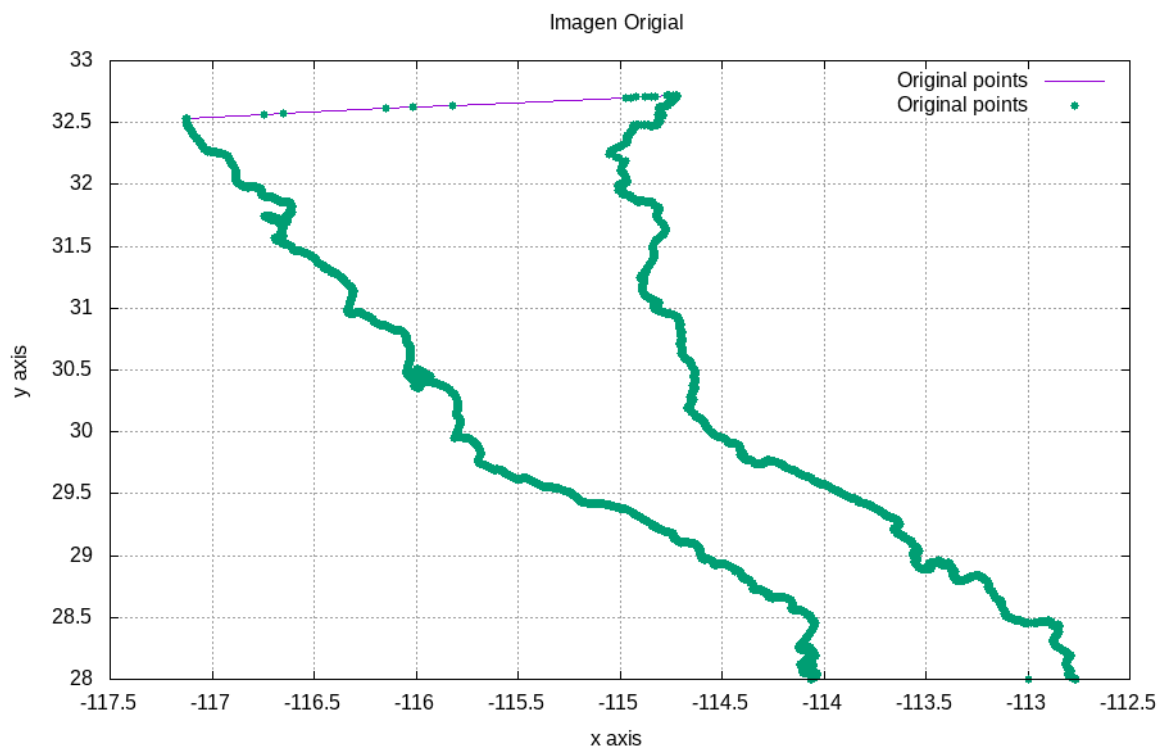


Figura 5. Imagen original con todos los vértices.

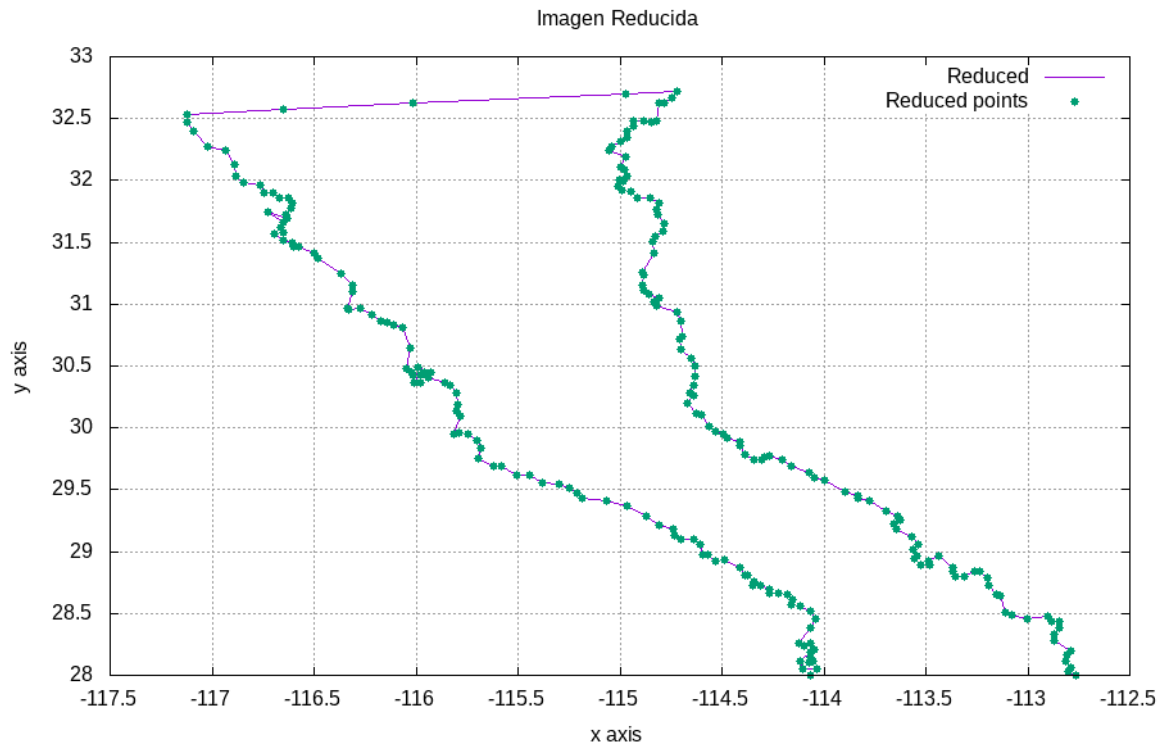


Figura 6. Imagen reducida.

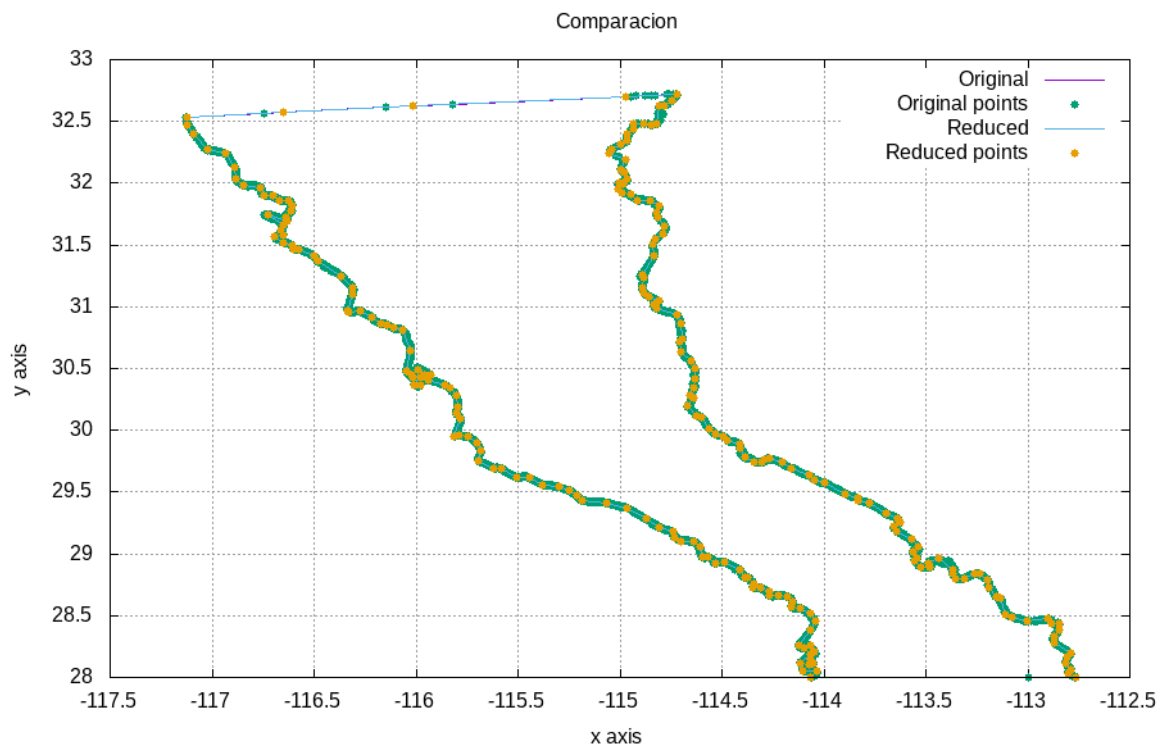


Figura 7. Comparación entre la imagen original y la imagen reducida.

Para este ejemplo las métricas de simplificación son:

- Vértices originales: 8715
- Vértices conservados: 229
- Porcentaje de compresión: 2.62 %
- Coeficiente de Jaccart: 1.0

**Tlaxcala**

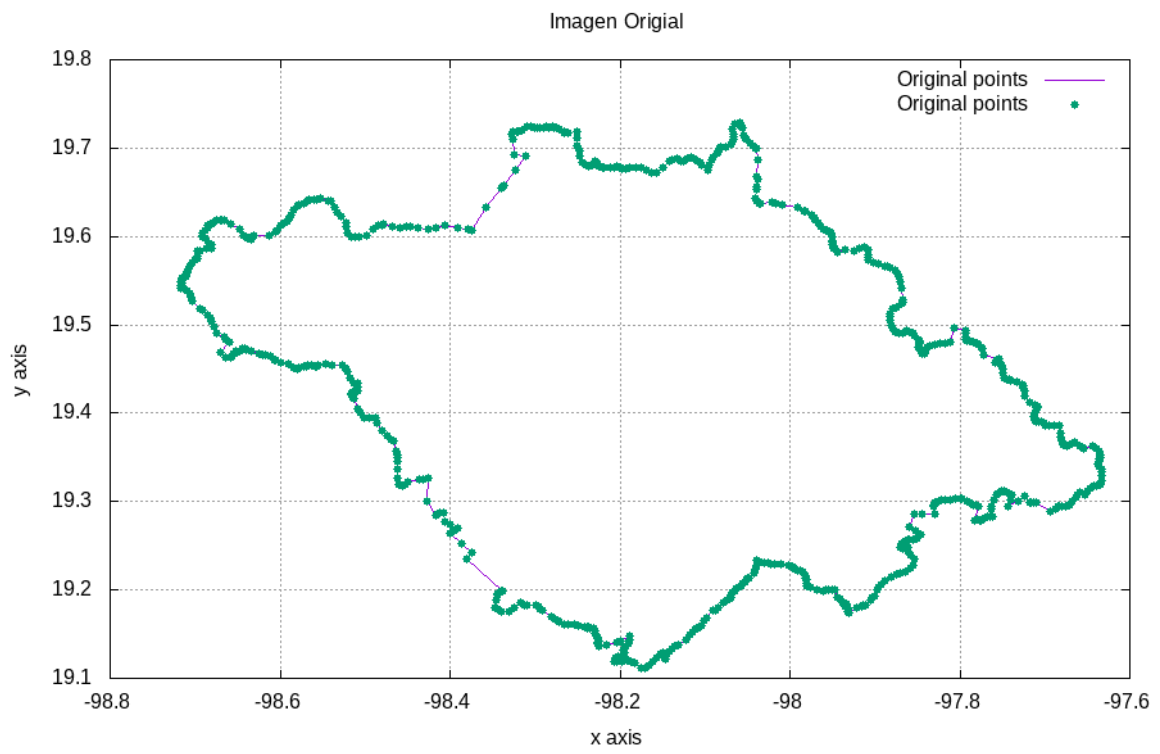


Figura 8. Imagen original con todos los vértices.

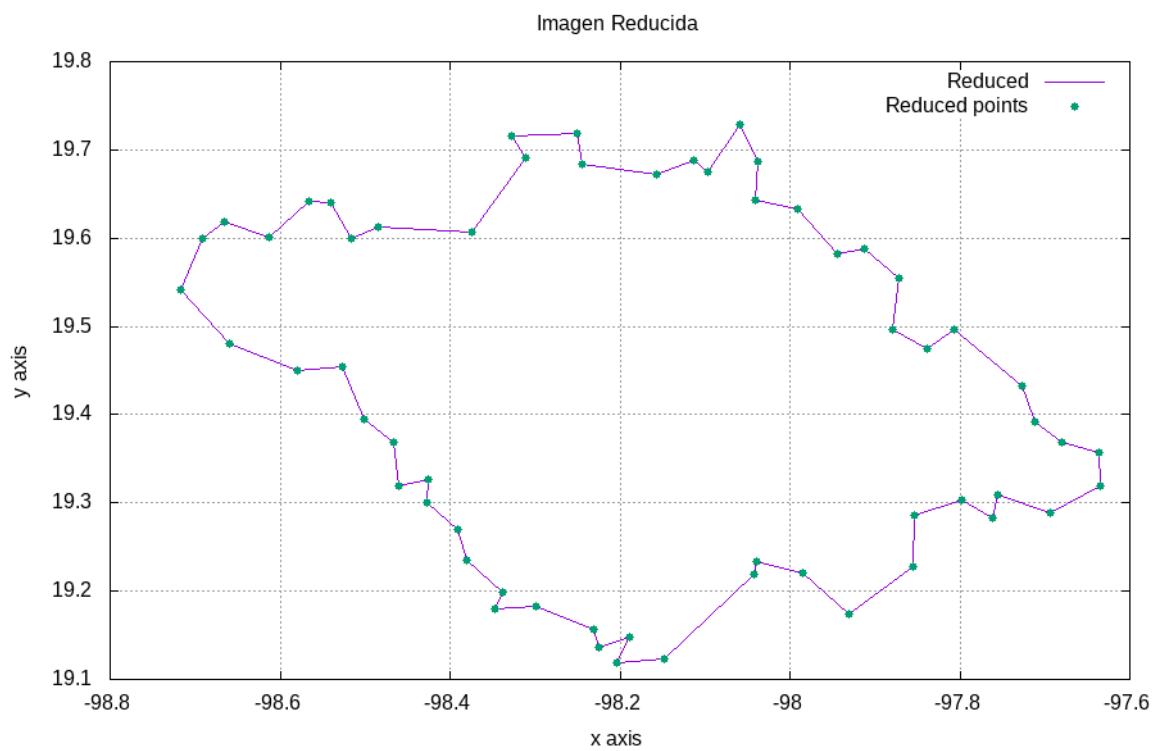


Figura 9. Imagen reducida.

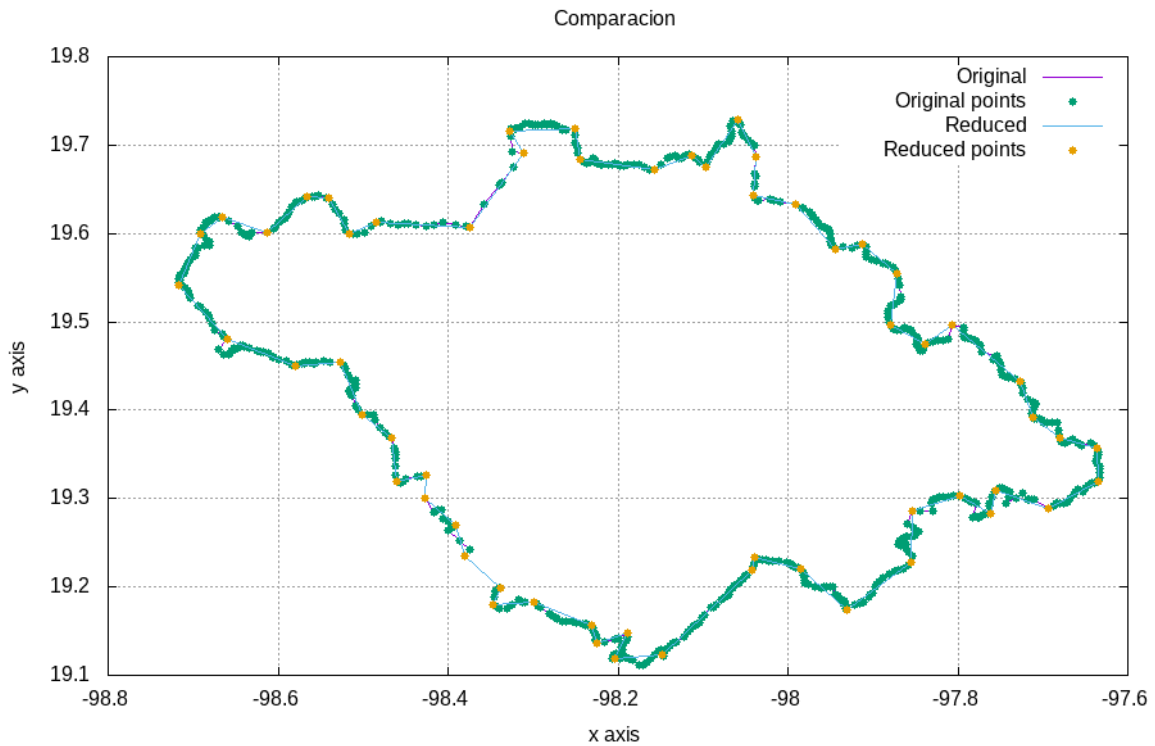


Figura 10. Comparación entre la imagen original y la imagen reducida.

Para este ejemplo las métricas de simplificación son:

- Vértices originales: 684
- Vértices conservados: 59
- Porcentaje de compresión: 8.62 %
- Coeficiente de Jaccart: 0.99

## Conclusiones

Se observa que con el error dado se tiene una buena visualización de los mapas con menos del 10% de los vértices originales (en el caso de Baja California se logra una compresión de menos del 3%) mientras se conserva la similitud de las formas.

Se utilizaron dos versiones de programa, uno para graficar en GNU plot (a partir del cual son mostradas las gráficas en la sección anterior de este reporte) y otro para graficar en Cairo. Las instrucciones para compilar y correr cada una de las versiones es diferente y cada forma se explica en el *README* contenido en cada carpeta. Para el caso de la versión en Cairo se tuvo problemas al convertir las coordenadas de los vértices del polígono en coordenadas en pixeles dentro del espacio de trabajo (lienzo). La solución a este problema fue cambiar a valores positivos cada coordenada dentro de los archivos de texto aunque esto significó trabajar con un reflejo de las imágenes originales. En seguida se muestran las imágenes generadas en Cairo.

