

# Representación Circular de VisDB: Una Visualización Multidimensional

Juan Diego Bejarano, 2017079378

jbejarano@ic-itcr.ac.cr

Yuberth Elizondo, 2016055077

yubelizondo@estudiantec.cr

Visualización de Información

Proyecto 2

Descripción, Análisis y Desarrollo técnicas de  
Visualización de VisDB con Técnicas innovadoras

## Introducción

VisDB es una aplicación que fue creada hace varios años por la “Institute for Computer Science, University of Munich”. VisDB utiliza técnicas de visualización de datos, el cual nos deja ver una cantidad grande de datos, normalmente proveniente de Bases de Datos Gigantescas, así se puede ver de manera eficaz, en una pantalla los datos sin perder la profundidad de todos los datos, es decir es una herramienta que soporta la exploración de grandes Bases de datos utilizando sistemas visuales humanos, para analizar dichas bases de datos. La técnica se basa en el uso de píxeles con colores para representar la cantidad de datos sin perder los datos por ser representados en un espacio tan pequeño, es decir una vez utilizada la técnica, el usuario recibe una representación gráfica, fácil de entender de todos los datos.

En Bases de datos extremadamente grandes, con miles de datos, o hasta millones de datos, normalmente es un problema encontrar los datos, y cómo están relacionados entre ellos; en sistemas de búsqueda de datos convencionales, como la búsqueda SQL, hasta las personas mas experimentadas en una Base de Datos específica tienen problemas en encontrar información o más comúnmente encontrar la relación entre 2 o más datos. A través de los años se han hecho una cantidad considerable de acercamientos para mejorar la búsqueda de casos específicos en bases de datos, entre estos tenemos un ejemplo que consiste en hacer una interfaz gráfica para ver de manera más fácil los datos (ej. FLEX [Mot 90] o GRADI [KL 92]) o por otro lado tenemos las técnicas que intentan dar un dato aproximado a una búsqueda en específico. El problema de estas técnicas es que utilizan métodos de generalización, por lo que los resultados que nos dan no son cien por ciento confiables. A partir de estos esfuerzos que se hicieron antes, se creó un tipo de búsqueda, en el cual toda la base de datos se la gráfica, no solo se hace una interfaz gráfica que ayude a buscar (y no solo en consola como se hacía antes) Haciendo la visualización de datos relacionados son fáciles de ver y se ve de manera agradable los Datos, de aquí nace un nuevo tipo de Técnicas de Visualización como la siguiente.

En nuestra adaptación de VisDB, Basándonos en las ventas de un grupo de vendedores de Walmart realizamos la adaptación de VisDB, utilizamos una técnica el cual se basa en graficar los datos, a partir de un valor de relevancia (en nuestro caso el valor de relevancia puede ser el valor menor o mayor de una base de datos), se grafica en una manera espiral, siguiendo un patrón claro, luego el color se define por el id del vendedor o valor de las ventas concluidas.

La visualización se realizó en la Herramienta llamada Dioköl, la cual se basa en Lua y Processing.

## **Antecedentes**

Todo desarrollo tecnológico tiene una razón de ser, en el caso de VisDB, se desarrolló por el hecho que las bases de datos científicas y geográficas tienden a tener cantidades de datos extremadamente grandes, al tener tantos datos, aparecen desafíos como por ejemplo; el buscar datos significantes se vuelve extremadamente complicado por el gran volumen de datos presentes, los usuarios no saben exactamente lo que están buscando, y con los sistemas de búsqueda tradicionales no es posible especificar y buscar ideas no precisas, es decir ideas vagas o “borrosas”, por lo que no es posible conseguir datos aproximados. Finalmente tenemos que en lo tradicional, no existe la retroalimentación, lo que conlleva a que los resultados pueden ser demasiados o muy pocos.

Lo anterior nos lleva a los requerimientos para poder armar un buen sistema de visualización y búsqueda para bases de datos grandes, estos son:

- Una manera interactiva de búsqueda y visualizar información.
- Una buena retroalimentación al usuario por medio de imágenes o visuales agradables a la vista.
- Los usuarios deben poder ver la mayor cantidad de datos posibles para poder ver clusters de información o patrones.
- Desplegar interdependencias entre los datos, es decir los datos relacionados

La idea básica de VisDB es visualizar los datos al mapear las distancias de los datos por medio de colores para representar cada uno de los items existentes de manera interactiva, por lo que podemos decir que lo que se quiere lograr con VisDB es crear una manera de visualizar resultados eficiente y sin perder profundidad de datos.

Entre las herramientas que VisDB posee es una mayor realimentación en los resultados que las búsquedas regresan, también la interactividad de las técnicas nos dejan que exista una retroalimentación al usuario inmediata al modificar la búsqueda, Por otro lado VisDB es una herramienta configurable, es decir, esta deja representar varios tipos o formas de datos con sus técnicas de visualización, y nos permite el uso de la visión humana para reconocer patrones en los datos presentes.

El enfoque que usa VisDB es utilizar todos los píxeles para visualizar los resultados, ya sea por color o frecuencia, y proveer datos, aunque sea una respuesta parcial, es decir no es exactamente lo que se buscó. Es decir, los resultados se aproximan por medio del factor de relevancia, dicho factor de un dato es obtenido al calcular distancias para cada sección de datos y combinarlas, entre menor distancia, mayor el factor de relevancia.

## **Justificación de la modificación a VisDB**

A partir de los datos que se han recolectado con el uso de las técnicas básicas hemos llegado a la conclusión de que como para relacionar los datos se utilizan el resultado de todos los factores relacionados, se separan las ventanas para mostrar de manera natural la búsqueda y sus resultados; pero lo que lleva esta división es que la visualización no permite una forma agradable a la vista humana y normalmente es confuso, por lo que la utilización de círculos, en vez de cuadrados ya que la técnica es una espiral, es más natural para el programador y para el usuario final, ya que sigue una verdadera espiral sin tener quiebres en la forma para poder realizarla en un cuadro, esta espiral cuadrada que se utiliza en VisDB, la llamamos una espiral falsa o quebrada, ya que no sigue lo que esperamos de una espiral natural, ya que el ser Humano está acostumbrado a ver espirales en maneras de círculos, no de cuadrados, por lo que la realización de una espiral natural o verdadera con círculos puede ampliar

la técnica de Visualización y mejorar en la búsqueda de patrones y muestras de datos relacionados.

## Implementación del Algoritmo

El algoritmo utilizado fue encapsular los arreglos generados por medio de los algoritmos originales de VisDB y el modificado, es decir la espiral natural en el método de manejo de píxeles de Dioköl para optimizar el proceso de dibujo de las técnicas de visualización de VisDB, originales y modificadas, así permitiendo una interacción mucho más eficiente con las técnicas ya que debe no dibujar una gran cantidad de píxeles, si no simplemente dibuja una imagen completa, reduciendo de manera exponencial la cantidad de cálculos que tiene que hacer la técnica.

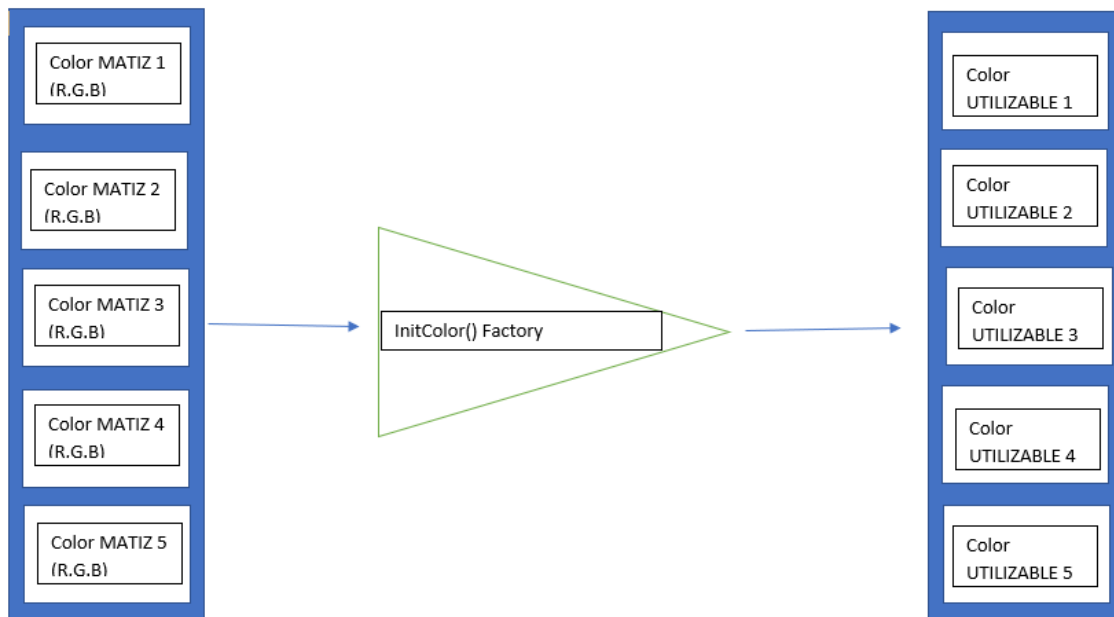
Para el uso de los colores, se basó en 5 colores base para la interacción, con 5 tonalidad de este mismo, los colores base fueron el rojo, verde, magenta, azul y café, utilizando el RGB (Red, Green y Blue) de los derivados en tablas para poder crear los colores que se pueden utilizar en los dibujos de las imágenes.

```
local azules={
  {0,255,255},
  {48,213,200},
  {135,206,255},
  {155,196,226},
  {127,255,212}
}

local cafes={
  {148,129,43},
  {204,119,34},
  {184,115,51},
  {218,138,95},
  {128,0,32}
}

function initColor(colores)
  local colors={}
  for i=1, #colores
  do
    local a=color(colores[i][1],colores[i][2],colores[i][3],255)
    table.insert(colors,a)
  end
  return colors
end
```

Aquí se presenta el ejemplo de los matices de colores base, con la función que genera el color utilizado en el pintado de imagen.



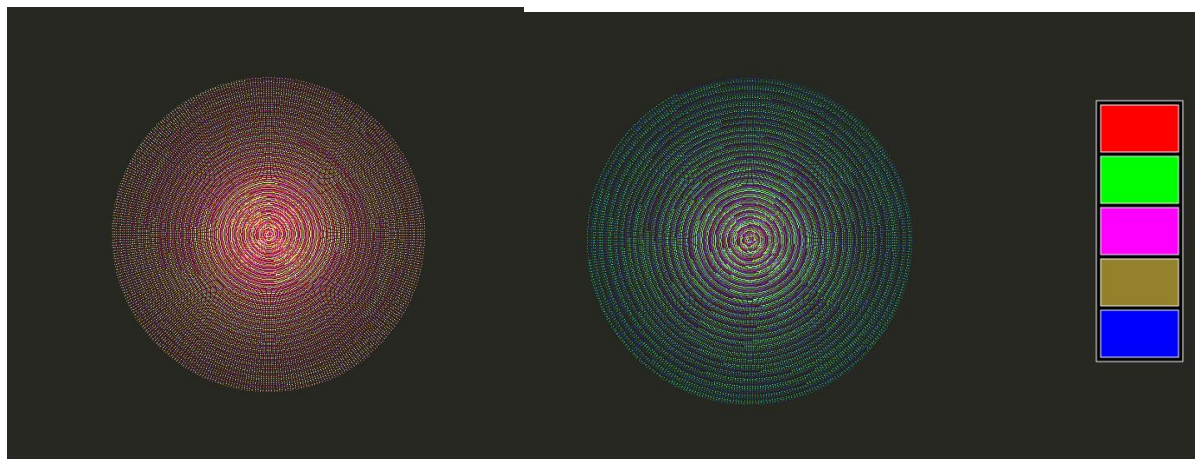
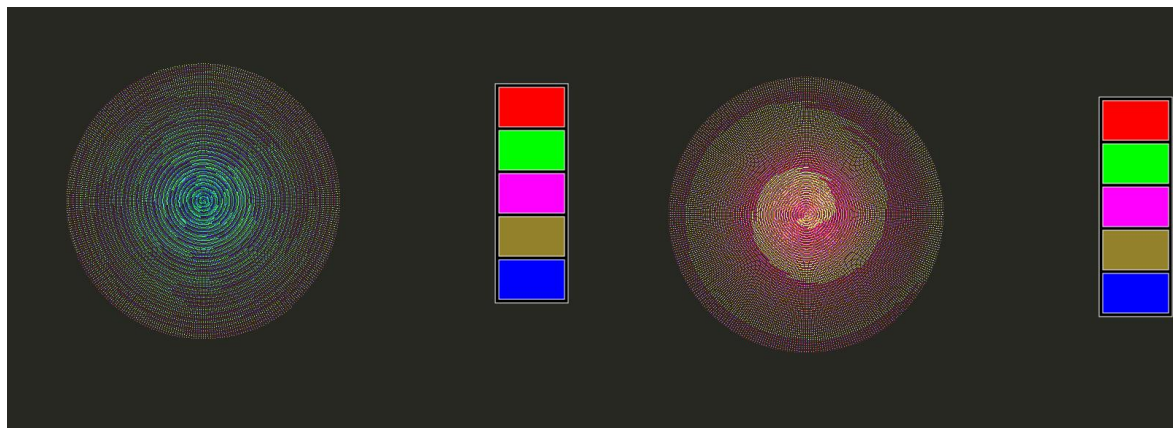
Este diagrama muestra cómo Init color pasa de RGB a un color utilizable por el programa.

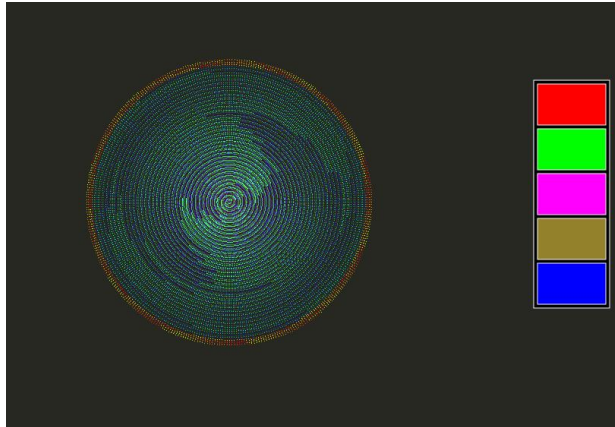
El algoritmo implementado para hacer la espiral natural, es el uso de radianes y senos y cosenos para calcular la posición x, y de cada uno de los píxeles, metiendolos luego en una tabla para luego meterlo en el array de píxeles para luego dibujarlo, en contraste a la espiral implementada por VisDB, la cual usa posiciones en un cuadrado para calcular el x, y del píxel individual, es decir, lo que hace VisDB es llenar un cuadrado de dimensiones que pueden llegar a ser infinitas, pero, el enfoque que nosotros utilizamos para la generación de nuestra técnica de espiral natural, hace que sea un círculo, con x, y más precisos que el cuadrado, haciendo que la visualización sea más real a la realidad, sin perder información.

La interacción se basa en un cambio de color de la imagen a partir de un eje en el lado derecho para poder utilizar los colores que uno desee en la visualización, entre los colores base, que son 5 bases, que cambian los colores a 5 tonalidades de dicha base.

## Evaluación

A partir del programa realizado, pudimos realizar pruebas en bases de datos de Walmart para poder visualizar los diferentes tipos de ventas realizadas, el uso de colores en la espiral, nos permite ver patrones entre los vendedores y las ventas específicas realizadas, por ejemplo, tenemos que entre más afuera de la espiral el ID del vendedor va creciendo y el color de cada uno de los píxeles de la espiral nos muestra el tamaño de la venta, por lo que dependiendo del color específico que se usa como base (Elegibles en el eje vertical de la derecha) se pueden apreciar diferentes patrones y datos más específicos si la gama de color elegida es más amplia.





Las imágenes anteriores muestran ejemplos de la visualización utilizada con alrededor de 40.000 datos, el uso de la espiral aceleró considerablemente la interacción de los datos y la creación de la visualización, ya que no hay un peso tan importante sobre el CPU en el ámbito de que no debe calcular tantos puntos de manera cuadrada, lo que lleva a que los cálculos sean demorados, más bien, al ser una espiral natural, el uso del CPU disminuye constantemente en comparación a las versiones anteriores de VisDB.

## Conclusiones

En general, lo que llegamos a hacer con esta técnica de visualización es la optimización de búsqueda de patrones en Bases de Datos considerables, al haber implementado un algoritmo tan liviano como la espiral natural, nos permitió reducir el tiempo de corrida de la aplicación, ya que no hay largos tiempos de espera mientras se realizan todos los cálculos (Que se realizaban anteriormente cada frame individual haciéndolo extremadamente pesado para el CPU), por otro lado la Interacción de la aplicación fue mejorado en eficiencia considerablemente, con resultados mostrados en pantalla prácticamente de manera instantánea, sin embargo una desventaja que tiene la espiral natural en vez de la espiral cuadrada es el tamaño requerido para poder mostrar la misma cantidad de datos, esto se da por el hecho que la espiral circular necesita más espacio para poder ser dibujada sin que los datos se pierdan, cosa que puede llegar a ser problemática al insertar la técnica en una aplicación que incluya un storyboard de diversas técnicas relacionadas con un conjunto de datos compartido.



En puntos de mejora futuros podemos mejorar la estabilidad del programa y la búsqueda de datos de un pixel en específico, ya que esto puede llevar a tener más profundidad en la búsqueda de datos, siendo aún más precisos en la representación de grandes cantidades de información

## **Bibliografía**

Etemandpur, R., Linsen, L., Paiva, J., Crick, C., & Forbes, A. (2018). Choosing Visualization Techniques for Multidimensional Data Projection Tasks: A Guideline with Examples. Stillwater, OK, USA: Oklahoma State University.

Google Patents. (2018). Retrieved from <https://patents.google.com/patent/US9633105B2/en>

Keim, D., & Kriegel, H. (1994). VisDB: Database Exploration Using Multidimensional Visualization. München: Institute for Computer Science, University of Munich. Retrieved from <https://epub.ub.uni-muenchen.de/4129/1/06.pdf>

lua-users wiki: Home Page. (2018). Retrieved from <http://lua-users.org/wiki/>

Messaoud, R., Boussaid, O., & Rabaséda, S. (2007). Efficient Multidimensional Data Representations Based on Multiple Correspondence Analysis [Ebook]. Bron Cedex, France: Laboratory ERIC – University of Lyon.

Processing Foundation. (2018). Retrieved from <https://github.com/processing>