

2.3 Aspectos teóricos

2.3.1 Correspondencia entre la red vial y la representación gráfica.

Cuando se plantea el caso práctico de introducir los datos relativos a una ciudad en la base de datos del programa el primer problema que hay que resolver es la determinación de un sistema de representación que permita identificar unívocamente cada punto de la ciudad.

Lo primero que se puede hacer es recurrir a un sistema ya definido, por ejemplo, latitud y longitud. Este sistema tiene el problema de que no es práctico para zonas geográficas pequeñas como, por ejemplo, áreas metropolitanas, que es precisamente lo que se tratará en este documento. Por ello, se debe pensar en otro sistema de coordenadas.

Se debe elegir un sistema sencillo pero práctico a la vez. Esto llevó a la elección de un sistema de representación propio. Lo único que se necesita para su implementación es un mapa completo del área y escuadra y cartabón. También puede usarse cualquier software de dibujo si se dispone de un mapa en formato electrónico.

En primer lugar se trazará un rectángulo sobre el mapa de tamaño tal que toda el área que interesa representar esté incluida en él. Lo que se pretende construir es un sistema de representación cartesiano en dos dimensiones. Se toma uno cualquiera de los vértices del rectángulo como origen de coordenadas. Los lados que se cortan formando dicho vértice serán los ejes de abscisas y de ordenadas. En la figura 2.3.1.1 se representa dicho sistema de ejes, sin incluir los lados del rectángulo que no forman parte del mismo.

A continuación se debe observar el mapa para decidir qué unidades de medida se van a usar. De acuerdo con la escala del mapa y con las unidades de medida, habrá que marcar los valores de las coordenadas en cada uno de los ejes. Por ejemplo, si el mapa tiene una escala 1:1000 y se toma como unidad de medida el metro, se debe hacer una marca en cada uno de los ejes cada 1 mm a partir del origen. Estas marcas deben numerarse consecutivamente desde este origen. Hay que tener en cuenta que la unidad de medida que se elija limitará la precisión del sistema de representación.

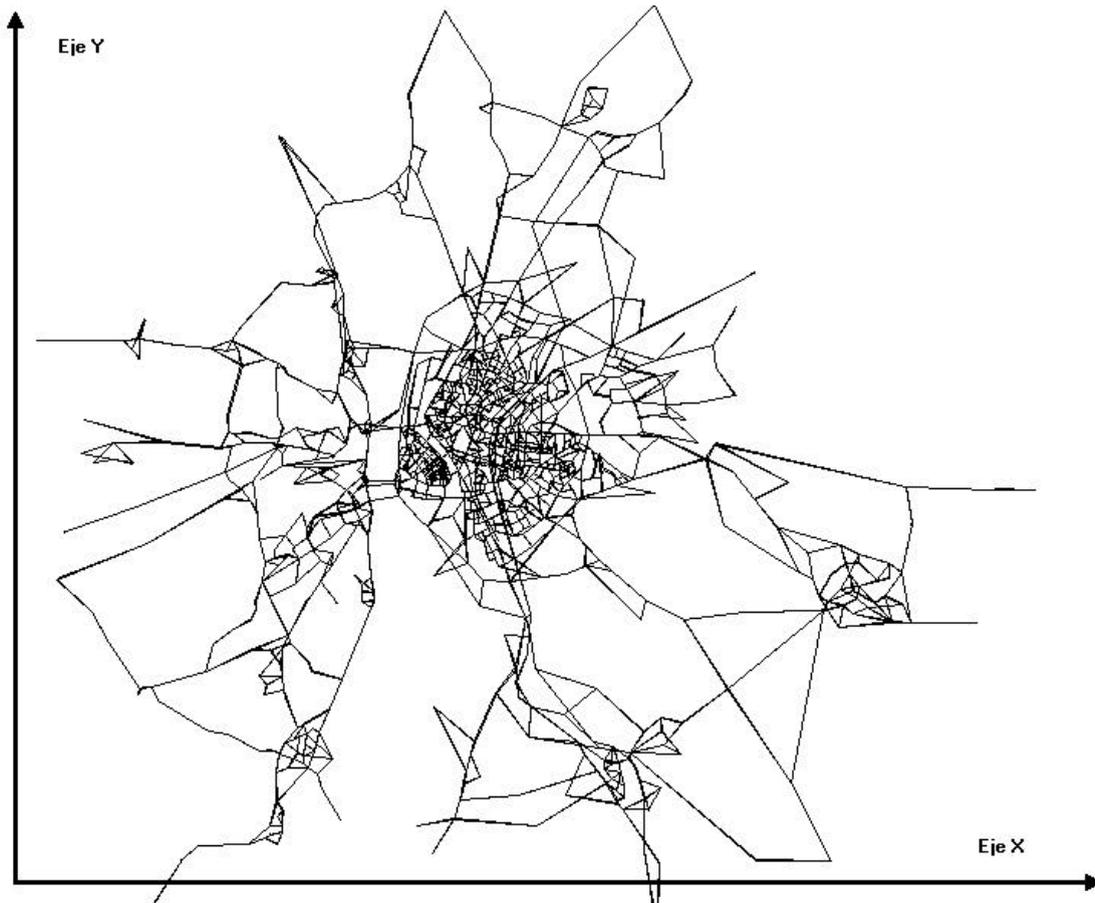


Figura 2.3.1.1: Sistema de representación global

Seguidamente se toma uno de los ejes y se trazan paralelas a él, hasta llegar al lado opuesto del rectángulo, trazando una paralela por cada marca, por cada unidad de medida, en el eje perpendicular al elegido. Una vez hecho esto, se toma el eje restante, perpendicular al anterior, y se procede análogamente, trazando paralelas a dicho eje hasta llegar al lado opuesto. De esta manera se tiene el área de interés totalmente cuadrículada. Ahora ya se puede decir que se ha construido un sistema cartesiano bidimensional de representación. En la figura 2.3.1.2 se muestra como han de quedar los ejes de coordenadas. En dicha figura se ha eliminado la red vial de la ciudad para una mejor apreciación, pero toda ella debía estar incluida en la zona cuadrículada.

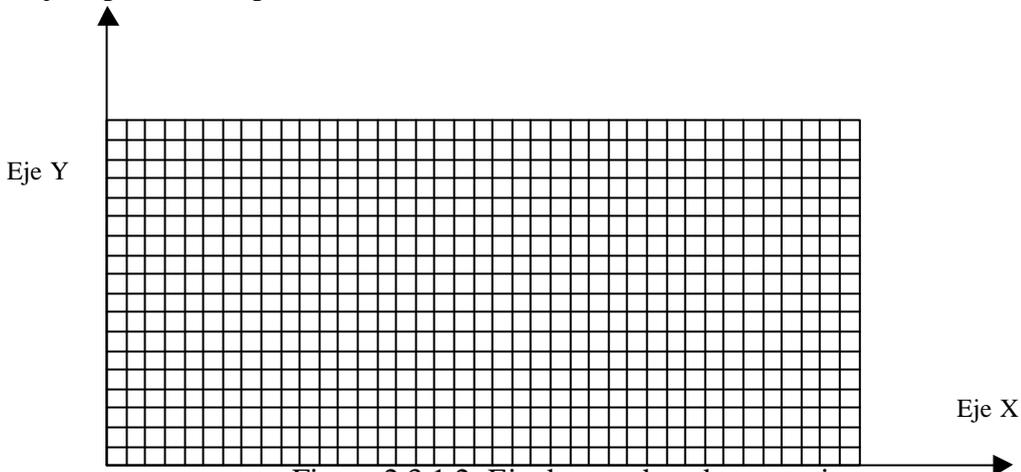


Figura 2.3.1.2: Eje de coordenadas cartesianas

Este rectángulo totalmente cuadrículado debe contener toda la red vial de la ciudad. A cada cruce de calles que aparezca en el mapa, que en lo sucesivo se denominarán nodos, se le asignan dos números, uno será la coordenada del eje horizontal y otro la coordenada del eje vertical. Esto se hace simplemente por aproximación, a cada nodo se le asignan las coordenadas del punto de corte más cercano a él. Un punto de corte no es más que una intersección entre dos líneas perpendiculares de las que aparecen en la figura 2.3.1.2.

En estos momentos se dispone de un sistema de representación que permitirá referirse a cada uno de los nodos de forma única. Este sistema tiene además la ventaja de que se puede elegir la precisión más conveniente. Los dos componentes básicos de la red vial de la ciudad son los nodos y los tramos. Como ya se visto, los nodos son los cruces de calles. Cada tramo representa un sentido de una calle. Es similar a un arco en un grafo dirigido cuyos nodos son los cruces. De esta manera, una calle por la que se puede circular en ambos sentidos estará representada por dos tramos, ya que cada tramo representa uno de los sentidos de circulación de dicha calle. Cada tramo puede identificarse y diferenciarse del resto conociendo su nodo origen y su nodo destino.

Pero aún queda la cuestión más importante, que es la representación gráfica de la red vial en una imagen, es decir, la representación en la pantalla del ordenador. A primera vista, parece un problema fácil de resolver: basta dibujar el mapa a una escala adecuada de manera que quede representado en su totalidad dentro de la imagen. Para no dar lugar a equívocos, hay que tener en cuenta que una imagen es un objeto que tendrá ciertas dimensiones y dentro del cual podremos representar gráficamente la red vial.

Pero el mapa no es sólo una representación gráfica de la imagen, sino que es un componente de la interfaz de usuario, es decir, debe ser capaz de interactuar con el usuario. El mapa debe ser capaz de determinar las coordenadas de un punto sobre la imagen en el sistema de representación de la ciudad.

Analizando el problema, se tiene que realmente hay dos sistemas de coordenadas distintos para describir el mismo conjunto de puntos. Uno de estos sistemas es el que se ha usado para identificar los nodos de la ciudad, el que se creó “con escuadra y cartabón”. A este sistema se le llamará sistema global o absoluto de coordenadas a partir de ahora. El motivo es que las coordenadas de un punto cualquiera en este sistema de coordenadas son constantes. El otro sistema de representación es el que se usa para localizar a un punto de la imagen. La imagen tiene un sistema de representación que usa el píxel como unidad de medida. Así la imagen tendrá un cierto tamaño en píxeles. Se recuerda que la imagen es un objeto que se crea y por tanto puede tener distintas dimensiones. Tanto si se está visualizando una panorámica de toda la red vial como si se está analizando, gracias al zoom, un área menor, el tamaño del objeto imagen será el mismo, una vez decidido éste. Al sistema de coordenadas asociado a la imagen se le conocerá como sistema de coordenadas locales o relativas desde este momento. El término relativas viene por el hecho de que las coordenadas de un cierto nodo en este sistema varían según se haya realizado o no un zoom y de la zona en que hagamos dicho zoom, mientras que un punto cualquiera, independientemente del lugar donde aparezca representado en la pantalla, siempre tendrá las mismas coordenadas globales o absolutas.

Además hay que tener presente que en el sistema de representación de coordenadas locales el origen está en el vértice superior izquierdo en lugar de estar en el inferior, como ocurre en el sistema de coordenadas globales. Además el eje Y es creciente hacia arriba en el sistema de coordenadas globales pero creciente hacia abajo en el sistema de coordenadas relativas, como puede apreciarse en la figura 2.3.1.3:

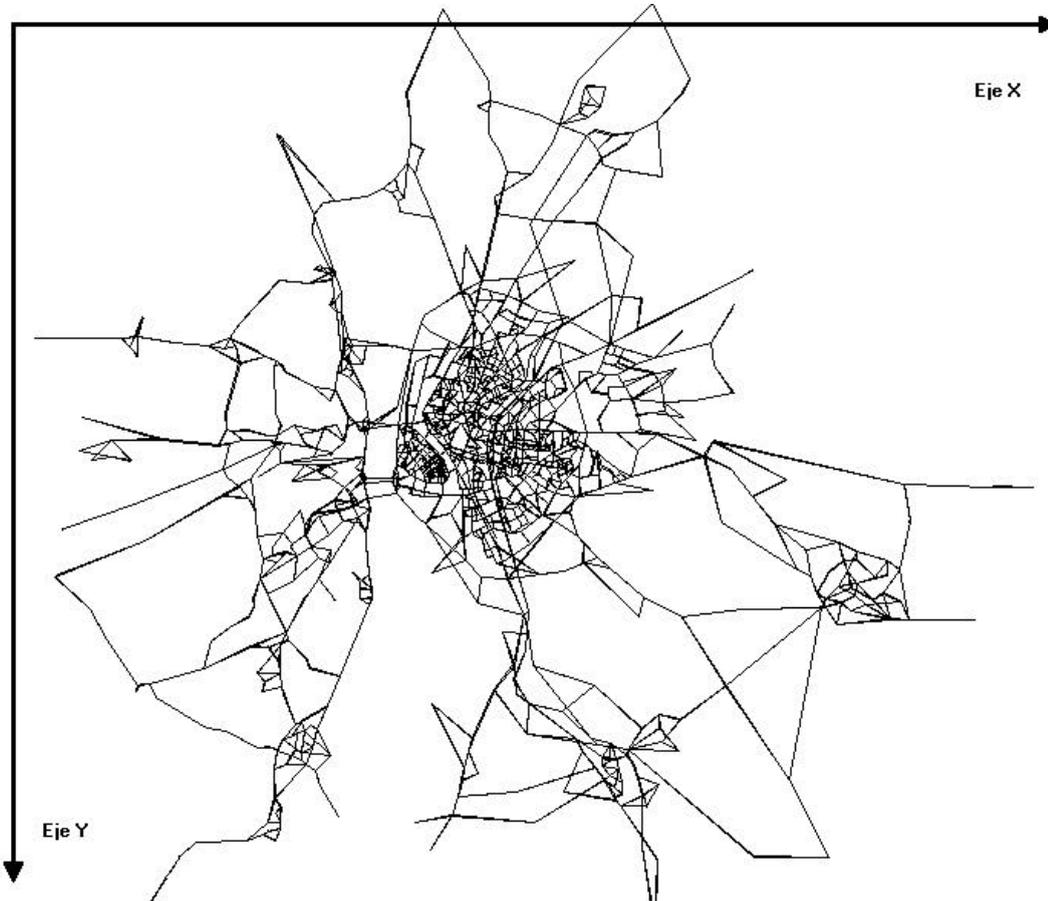


Figura 2.3.1.3: Sistema de representación local o relativo

La idea es que cuando el usuario seleccione un punto cualquiera de la imagen con el ratón, el sistema sea capaz de encontrar las coordenadas globales de ese punto, independientemente de si está visualizando la red vial al completo o sólo una porción de la misma y del tamaño con que aparezca en la pantalla. Por tanto, es necesario que exista una correspondencia unívoca entre ambos sistemas de coordenadas: a cada punto de la imagen le corresponde uno y sólo uno de los puntos del sistema global. Es decir, para cada imagen, conociendo ciertas condiciones del entorno, se tendrá una función de transformación con la que se obtendrán las coordenadas globales de cada uno de sus puntos, y viceversa. Desde ahora se denotará por f (*posición relativa, variables entorno*) a la función que, dada la posición relativa de un punto de la imagen en píxeles, devuelve el valor de ese punto en coordenadas absolutas o globales. Y por supuesto la función f^{-1} (*posición absoluta, variables entorno*) hará la transformación opuesta, es decir, dado un punto cualquiera en coordenadas globales, devuelve la posición relativa de ese punto en la imagen que actualmente se muestra en pantalla, en coordenadas relativas (píxeles).

Así, el problema queda limitado a definir las variables de entorno y hallar las funciones de transformación necesarias, teniendo presente que éstas quedarán expresadas en función de las variables de entorno.

Las variables de entorno que se usarán son:

- *límite*: esta variable indica la porción de ciudad que se representa en pantalla usando coordenadas globales. Esta variable consta en realidad de dos puntos, que definen un rectángulo. La porción de ciudad que está dentro de dicho rectángulo es la que se mostrará en pantalla. El primer punto es la esquina superior izquierda del rectángulo y el segundo es la esquina inferior derecha, con lo que los límites del rectángulo quedan definidos. La notación será $lx1$ y $ly1$ para las coordenadas x e y del primer punto y $lx2$ y $ly2$ para las coordenadas x e y del segundo punto, respectivamente.
- *entorno*: proporciona una relación de proporción entre el sistema de coordenadas globales y el relativo asociado a la imagen actual que muestra la pantalla. Para calcularlo, se divide el ancho de la imagen representada en pantalla (en píxeles) entre el ancho del rectángulo definido por la variable *límite* y el alto de dicha imagen entre el alto de dicho rectángulo. El valor válido es el menor de los dos. Es decir:

$$entorno1 = \frac{Ancho_imagen}{Ancho_rectángulo_límite} = \frac{Ancho_imagen}{lx2 - lx1}$$

$$entorno2 = \frac{Alto_imagen}{Alto_rectángulo_límite} = \frac{Alto_imagen}{ly1 - ly2}$$

$$entorno = \min \{entorno1, entorno2\}$$

En estas dos expresiones *Ancho_imagen* y *Alto_imagen* son coordenadas relativas y están, por tanto, expresadas en píxeles, mientras que *Ancho_rectángulo_límite* y *Alto_rectángulo_límite* son coordenadas globales y estarán expresadas en el sistema de medida que se haya elegido para este sistema de coordenadas.

- *desplazamiento*: en realidad son dos variables, desplazamiento de la coordenada x (*desplazamiento_x*) y desplazamiento de la coordenada y (*desplazamiento_y*). Esta variable tiene como único fin que la red vial aparezca representada centrada en la imagen. Es necesario, por tanto, calcular su valor antes de representar cualquier imagen. Para calcularlas lo que se hace es obtener el centro de la imagen y el centro del rectángulo definido por *límite*, que define la porción de ciudad que se representa en la imagen. Matemáticamente:

$$centro_imagen_x = \frac{f(Ancho_imagen, entorno, limite) + f(0, entorno, limite)}{2}$$

$$centro_imagen_y = \frac{f(Alto_imagen, entorno, limite) + f(0, entorno, limite)}{2}$$

$$centro_rectángulo_limite_x = \frac{lx2 + lx1}{2}$$

$$centro_rectángulo_limite_y = \frac{ly2 + ly1}{2}$$

$$desplazamiento_y = centro_imagen_y - centro_rectángulo_limite_y$$

$$desplazamiento_x = centro_imagen_x - centro_rectángulo_limite_x$$

Lo que se hace para obtener el centro tanto de la imagen como del rectángulo definido por *límite* es hallar las coordenadas máximas y mínimas para cada uno de ellos y hacer la media. El valor mínimo, tanto en el eje x como en el y, de la imagen en coordenadas relativas es 0. El valor máximo en coordenadas relativas para el eje x será el ancho de la imagen y el valor máximo para el eje y será la altura. En el caso de la imagen estos valores deben convertirse a coordenadas globales. A continuación se puede calcular el centro de la imagen como se ha comentado, y lo obtendremos en coordenadas globales, pues anteriormente se ha realizado la transformación.

Para hallar el centro del rectángulo definido por *límite* las coordenadas que tenemos ya son globales, por lo que no hay que hacer ninguna transformación, sino que directamente se calcula el punto medio para obtener el centro. Una vez obtenidas las coordenadas x e y de los centros de tanto la imagen como el rectángulo definido por *límite* no hay más que hacer una sencilla sustracción para obtener el valor del desplazamiento que se tendrá que usar para que la porción de red vial a representar aparezca centrada en la imagen. Hay que dejar claro que la imagen es una zona de pantalla donde aparecerá representada gráficamente una porción de la red vial, quedando ésta última definida por la transformación del rectángulo límite a coordenadas relativas y que aparecerá representada dentro del espacio reservado para la imagen aunque quizás no ocupando todo él. De aquí la necesidad de las variables de entorno *desplazamiento_x* y *desplazamiento_y*. Esto queda aclarado con la figura 2.3.1.4.

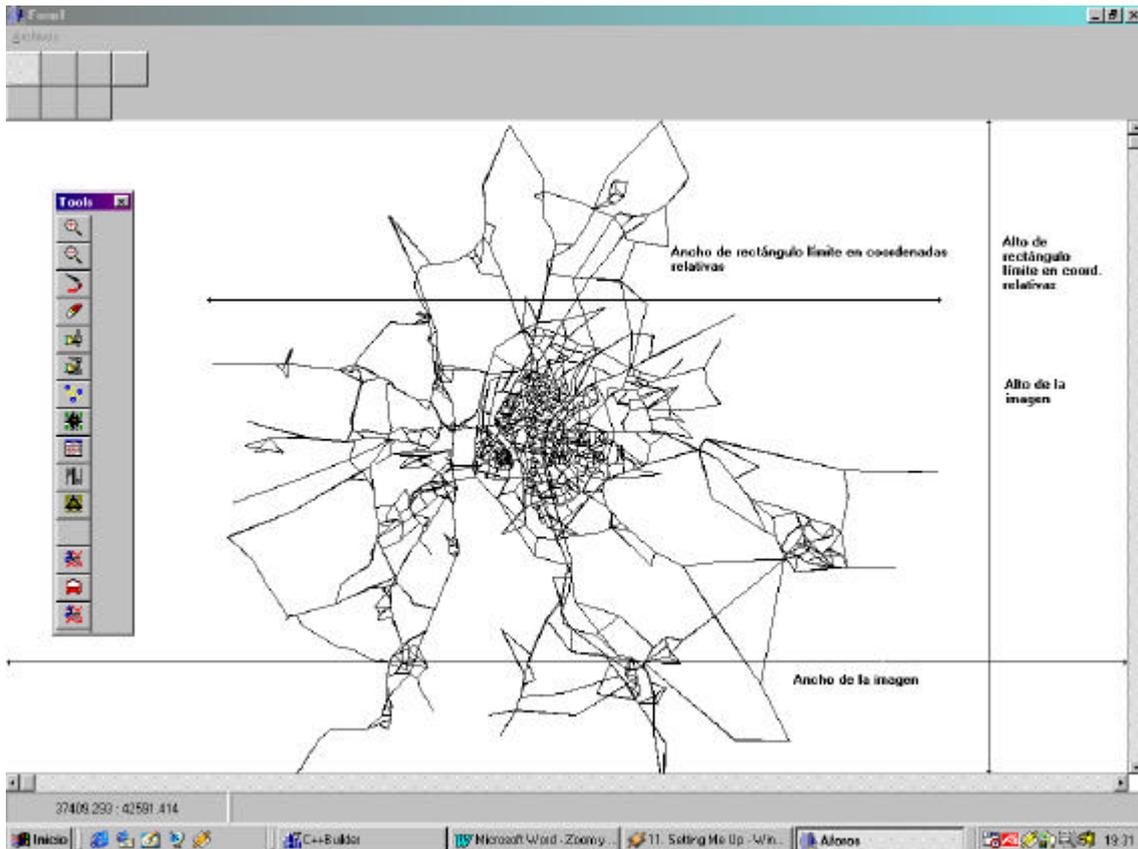


Figura 2.3.1.4: Imagen frente a rectángulo límite en coordenadas relativas

Si no se hubiesen definido *desplazamiento_x* y *desplazamiento_y*, la imagen obtenida sería como la que se observa en la figura 2.3.1.4 pero desplazada completamente hacia la izquierda. En dicha figura puede observarse como la red vial aparece representada en el centro de la imagen.

Queda por definir la función de transformación f (*posición relativa, variables entorno*) que, como ya se ha dicho, debe obtener las coordenadas globales de un punto al seleccionarlo en la imagen, gracias a las variables de entorno. En definitiva, transformar unos valores *coordenada_x* y *coordenada_y*, que serán proporcionados por el sistema al pinchar sobre un punto de la imagen y que estarán en píxeles (relativos a la imagen), a coordenadas globales. El algoritmo sería algo así:

- 1- Dividir *coordenada_x* y *coordenada_y* entre el valor de la variable *entorno*. En el caso de la *coordenada_y*, hay que tener en cuenta que los píxeles de la imagen tienen un sistema de ejes distinto al de las coordenadas globales. Esto se debe a que en una imagen, el origen está en la esquina superior izquierda, mientras que en el sistema de coordenadas globales este origen está en la esquina inferior izquierda. Por ello, en el sistema de coordenadas locales, el eje y es creciente hacia abajo, mientras que en el sistema de coordenadas globales es creciente hacia arriba. Por ello, en el caso de la coordenada y , lo que se debe dividir es *Alto de la imagen(en píxeles)-coordenada_y*. Tras realizar este primer paso se tiene la imagen en un sistema de coordenadas igual al de coordenadas globales pero desplazado respecto a éste.

2- Para corregir el desplazamiento, al resultado anterior obtenido para *coordenada_x* hay que sumarle el valor de la coordenada x del vértice superior izquierdo del rectángulo de la variable *límite*. Análogamente, al resultado obtenido para la *coordenada_y* hay que sumarle el valor de la coordenada y del vértice inferior derecho del rectángulo de la variable *límite*. Esto tiene como fin corregir el desplazamiento.

A continuación se puede observar la representación matemática del algoritmo que implementa *f* (*posición relativa, variables entorno*), teniendo presente que *coordenada_x* y *coordenada_y* son las coordenadas de un punto cualquiera en el sistema de coordenadas relativas, es decir, se expresa en píxeles desde el origen de la imagen. Se denotarán como *coordenada_x_global* y *coordenada_y_global* las coordenadas de este punto en el sistema de coordenadas global:

$$coordenada_x_global = \frac{coordenada_x}{entorno} + lx1$$

$$coordenada_y_global = \frac{(Alto_imagen - coordenada_y)}{entorno} + ly1$$

Recordar que *Alto_imagen* no es más que el alto de la imagen en píxeles y que *lx1* y *ly1* eran las coordenadas x e y del vértice superior izquierdo del rectángulo definido por *límite*. La variable *entorno* es la variable de entorno definida anteriormente. Queda por tanto definida la función de transformación *f*(*posición relativa, variables entorno*) y *f*⁻¹(*posición relativa, variables entorno*), ya que los valores de *coordenada_x* y *coordenada_y* pueden despejarse fácilmente de las expresiones anteriores:

$$coordenada_x = entorno \times coordenada_x_global - lx1$$

$$coordenada_y = entorno \times coordenada_y_global - ly1$$

Debe notarse como en el algoritmo de transformación de coordenadas no aparece en ningún momento la variable de entorno *desplazamiento_x* y *desplazamiento_y*. En efecto, esta función de transformación debe aplicarse una vez eliminado el desplazamiento que introducen las variables de entorno *desplazamiento_x* y *desplazamiento_y*, ya que éste desplazamiento no es tenido en cuenta por la transformación. Esto se verá más claro cuando se presente el procedimiento para dibujar en la imagen y los algoritmos de zoom y windowing. En resumen, que una función de transformación completa debería incluir también la corrección de *desplazamiento_x* y *desplazamiento_y*.

A continuación se presentará el procedimiento general para representar la porción de red vial delimitada por el rectángulo *límite* en la imagen. Es el último paso que resta para completar la representación básica de la imagen. Una vez detallado, se presentarán los algoritmos necesarios para implementar las funciones de zoom, usada para aumentar la resolución de la imagen, y su contraria, que se denominará windowing.

La información básica necesaria para poder representar gráficamente consta de los nodos y los tramos. Se denota por nodo el punto de confluencia de varias calles o vías en general, mientras que tramo designa cada uno de los sentidos de cada una de estas calles o vías. El proceso de dibujo de la red vial se ha dividido en dos etapas: dibujo de tramos y dibujo de nodos.

Primero tiene lugar el dibujo de los tramos. Para ello se debe obtener la posición de cada tramo que, por supuesto, estará almacenada en coordenadas globales. La posición de un tramo vendrá determinada por la posición de su nodo origen y su nodo destino. Por lo tanto se debe conocer el nodo origen y destino de cada tramo.

A cada coordenada de cada uno de los nodos extremos del tramo se le suma el desplazamiento calculado para la imagen a representar. Esto tiene como objetivo que cuando a continuación se le aplique la función f^{-1} (*posición relativa, variables entorno*) se obtengan las coordenadas relativas donde están los extremos del tramo realmente, teniendo ya la representación gráfica centrada en la imagen.

Para dibujar un tramo se trazará una línea desde el nodo origen al destino. Aunque todavía no se ha explicado el método de representación de los nodos, en la figura 2.3.1.5 se supone que son circunferencias:



Figura 2.3.1.5: Tramo y nodos

En la figura 2.3.1.5 se muestra una calle de doble sentido que por tanto está representada por dos tramos, uno en cada sentido. Hay que tener en cuenta que si se dibujan los tramos como líneas que unen los puntos correspondientes a los nodos origen y destino se tendría que el tramo en un sentido se superpondría con el tramo en el sentido opuesto. Por ello es necesaria una artimaña como la que se muestra en la figura 2.3.1.5: se supone que el nodo es una circunferencia de radio mayor que cero, no un punto. De esta manera se pueden representar los dos tramos, uno para cada sentido, sin que se superpongan. Por lo tanto no basta con transformar las coordenadas globales de los nodos a relativas, sino que hay que determinar el punto de origen y destino de cada una de las líneas que representan un tramo.

Pero esto hace que aparezca otro pequeño inconveniente. En efecto, en la figura 2.3.1.5 se representa una calle de doble sentido (dos tramos) que forma cero grados con la horizontal. Si se intentan representar tramos que formen un ángulo pronunciado con la horizontal y no se modifica el punto del que parten las líneas paralelas en la anterior figura, se puede dar un caso como el que se representa en la figura 2.3.1.6.

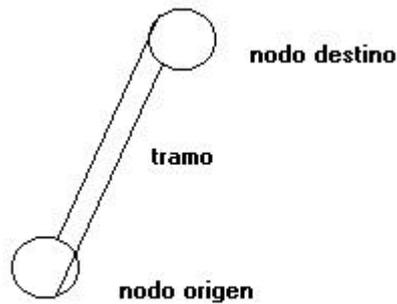


Figura 2.3.1.6: Tramo no horizontal y nodos

Se debe intentar que la imagen no presente estos defectos.

Para ello, hay que introducir una corrección en el cálculo de las coordenadas origen y destino de cada una de estas dos líneas, que no pueden ser siempre las mismas, sino que van a depender del ángulo que forma el tramo a representar con la horizontal. Esto hace necesario, en primer lugar, el cálculo de este ángulo y a continuación el cálculo de las coordenadas apropiadas de inicio y fin de cada una de las líneas paralelas.

Calcular el ángulo es sencillo conociendo las coordenadas globales de los nodos y teniendo en cuenta la figura 2.3.1.7.

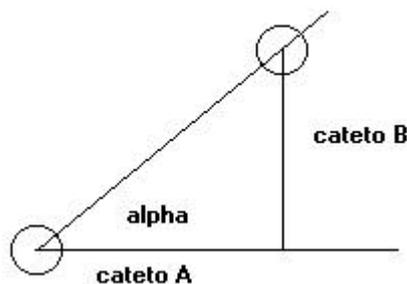


Figura 2.3.1.7: Ángulo a calcular

El proceso para calcular el ángulo *alpha* que nos interesa es:

$$\tan(\alpha) = \frac{\text{catetoB}}{\text{catetoA}}$$

De donde se obtiene:

$$\alpha = \tan^{-1}\left(\frac{\text{catetoB}}{\text{catetoA}}\right)$$

Que se puede evaluar sabiendo que *cateto B* es la diferencia entre la coordenada *y* del punto destino y la coordenada *y* del punto origen y análogamente *cateto A* es la diferencia entre la coordenada *x* del punto destino y la coordenada *x* del punto origen. El ángulo calculado es el formado por la línea recta que une las posiciones de los nodos origen y destino. Los puntos de inicio y origen vendrán determinados por un desplazamiento tanto en las coordenadas *x* como *y*, que se denotará a partir de ahora como *despl_x* y *despl_y*. Este desplazamiento sirve para calcular el punto origen y destino de cada una de las líneas que representan un tramo: *despl_x* indica desplazamiento de la coordenada *x* respecto a la posición del nodo (ya sea origen y destino) y *despl_y* indica el desplazamiento de la coordenada *y* respecto a la posición del nodo. La figura 2.3.1.8 servirá para aclarar este punto.

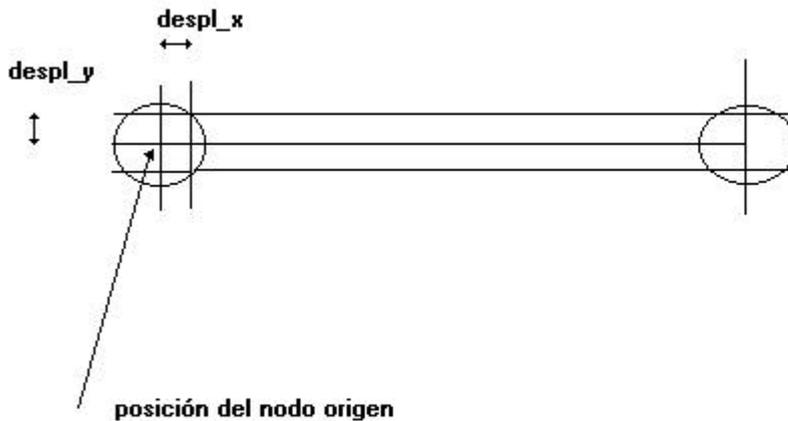


Figura 2.3.1.8: Desplazamientos en tramos horizontales

Lo normal es que *despl_x* sea cero y *despl_y* coincida con un radio imaginario para el nodo. Hay que tener en cuenta que la circunferencia que se está usando para representar el nodo tiene únicamente como fin facilitar el dibujo de las dos líneas paralelas que unen el nodo origen y destino, ya que los nodos se dibujarán a continuación, y ni tienen por qué tener ese tamaño ni esa forma. Lo que sí es conveniente, y es lo que se ha hecho, es elegir el radio de dicha circunferencia, o sea, *despl_y*, en función de la variable *entorno*, para evitar que se vea demasiado grande si se está visualizando la red vial completa y demasiado pequeña si se está visualizando una parte pequeña de la misma. Lo que quiere decir que *despl_y* se elija en función de la variable *entorno* es que su valor sea una constante multiplicada por la variable de *entorno*, de manera que mantenga la proporción independientemente de la resolución de la imagen.

Una vez decididos los valores de *despl_x* y *despl_y* lo que hay que hacer es aplicar la rotación del ángulo de inclinación del tramo respecto a la horizontal, como se muestra en la figura 2.3.1.9.

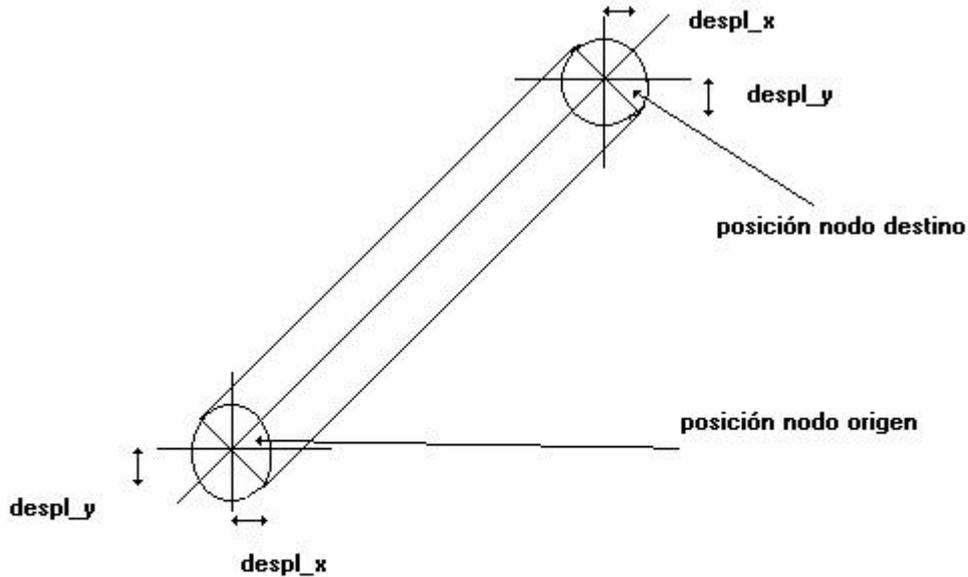


Figura 2.3.1.9: Desplazamientos con rotación

En la figura 2.3.1.9 se están representando las cotas, no se está teniendo en cuenta el signo.

Matemáticamente puede plasmarse:

Partiendo de las coordenadas globales de los nodos origen y destino, x_1-y_1 y x_2-y_2 , respectivamente:

$$x1_despl = x1 + desplazamiento_x$$

$$y1_despl = y1 + desplazamiento_y$$

$$x2_despl = x2 + desplazamiento_x$$

$$y2_despl = y2 + desplazamiento_y$$

Lo primero que se ha hecho es corregir el desplazamiento. A continuación se deberá aplicar la rotación a $despl_x$ y $despl_y$. Esto es fácil teniendo en cuenta que la línea que une el punto de inicio de la línea que representa el tramo con el centro de la circunferencia que representa al nodo forma con el eje x un ángulo igual al ángulo de inclinación del tramo y con el eje y un ángulo que es el complementario del anterior. Por lo tanto, es simplemente una rotación de ejes:

$$Despl_x_rot = despl_x \times \cos(\alpha) + despl_y \times \sen(\alpha)$$

$$Despl_y_rot = despl_y \times \cos(\alpha) - despl_x \times \sen(\alpha)$$

Ahora sólo resta calcular las coordenadas de origen y destino para cada una de las líneas. Es simplemente añadirles el desplazamiento:

$$\text{Coordenada_x1} = (X1_despl - lx1) \times entorno + Despl_x_rot$$

$$\text{Coordenada_y1} = Alto - (Y1_despl - ly1) \times entorno + Despl_y_rot$$

$$\text{Coordenada_x2} = (X2_despl - lx1) \times entorno + Despl_x_rot$$

$$\text{Coordenada_y2} = Alto - (Y2_despl - ly1) \times entorno + Despl_y_rot$$

Lo que se ha hecho es calcular las coordenadas relativas y sumarles el desplazamiento. Como el desplazamiento es del tipo constante* *entorno*, guardará la proporción de acuerdo con la variable *entorno*.

La línea tiene como punto de origen (*Coordenada_x1*, *Coordenada_y1*) y como punto final (*Coordenada_x2*, *Coordenada_y2*).

El paso siguiente es dibujar los nodos, que es bastante más sencillo. Lo que se hace es elegir unas nuevas variables *despl_x* y *despl_y*, que de nuevo dependerán de la variable *entorno* si se desea su correcta visualización en cualquier momento.

Cada nodo se representa como un pequeño rectángulo definido por *Coordenada_x1*, *Coordenada_y1*, *Coordenada_x2*, *Coordenada_y2*, de manera que el vértice superior del rectángulo es el punto (*Coordenada_x1*, *Coordenada_y1*) y el vértice inferior derecho es el punto (*Coordenada_x2*, *Coordenada_y2*). El número 10 que aparece en las expresiones es la mitad del lado del rectángulo, que al estar multiplicando por *entorno* es como si estuviese en coordenadas globales, es decir, que se adaptará a la resolución de la imagen en ese momento, al igual que se hizo anteriormente con los tramos. Si la mitad del lado del rectángulo es mayor que el radio de la circunferencia imaginaria que se definió para representar los tramos, los nodos y tramos aparecerán correctamente representados como se aprecia en la figura 2.3.1.10, ya que la representación del nodo ocultará los puntos de origen y fin de cada uno de los tramos. En caso contrario, en la imagen se apreciará que los tramos no llegan a conectar completamente con los nodos.

$$\text{Coordenada_x1} = (X - 10 + desplazamiento_x - lx1) \times entorno$$

$$\text{Coordenada_x2} = (X + 10 + desplazamiento_x - lx1) \times entorno$$

$$\text{Coordenada_y1} = Alto - (Y - 10 + desplazamiento_y - ly1) \times entorno$$

$$\text{Coordenada_y2} = Alto - (Y + 10 + desplazamiento_y - ly1) \times entorno$$

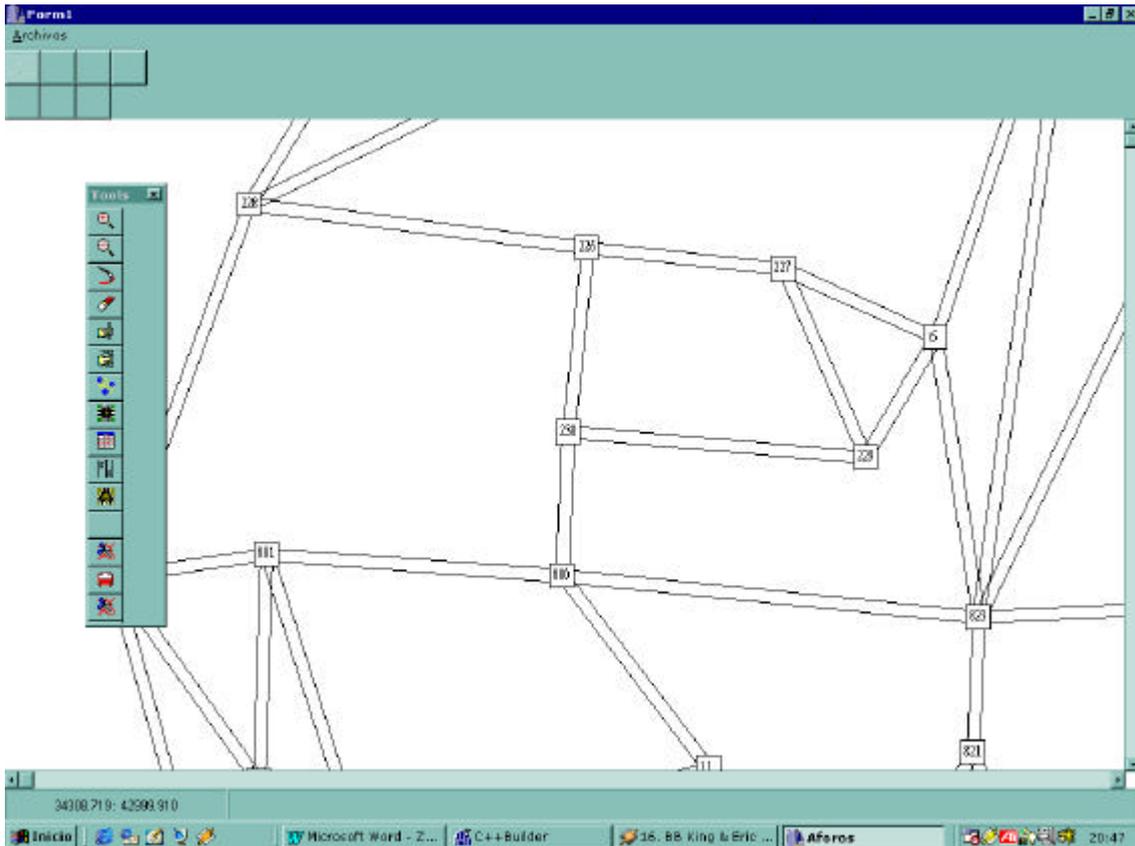


Figura 2.3.1.10: Imagen en la que se aprecian los tramos y nodos

En realidad cada vez que se redibuja la imagen se está pintando toda la red, no sólo la parte que aparece en ella. Esto es debido a que es posible que haya dos nodos que estén fuera de la porción de red vial a representar, pero que el tramo que los une sí esté dentro de dichos límites. Existen algoritmos para calcular qué tramos se deben dibujar y cuáles no en cada caso, pero no aportaban una mejora en cuanto tiempo, es decir, el tiempo en procesar dichos algoritmos es del mismo orden del que se tarda en dibujar toda la imagen e incluso superior. La solución por la que se ha optado es que, simplemente, los nodos, tramos y partes de los tramos que quedan fuera de la imagen no se ven representados.

A continuación se verá detalladamente la realización del zoom y del windowing. Cualquiera puede fácilmente hacerse una idea de qué es un zoom, en este documento se denotará como zoom la acción de ampliar, es decir, representar con mayor resolución, una porción seleccionada de una imagen. Por windowing se entenderá el proceso contrario, es decir, reducir la resolución.

En primer lugar se detallará el proceso de zoom. Para realizarlo correctamente se deben seguir los siguientes pasos:

En primer lugar, determinar el área que ha seleccionado el usuario para ampliar. Los datos que se usan para determinar el área elegida son las coordenadas del vértice superior izquierdo e inferior derecho del rectángulo que ha seleccionado el usuario.

A continuación se deben convertir los vértices anteriores en coordenadas globales. Por supuesto que para ello se deben usar las variables *entorno* y *límite*.

A estos vértices se les debe restar el *desplazamiento*, ya que éste se les sumó a la hora de representar la imagen en pantalla para obtenerla centrada. En este momento se dispone de los vértices del rectángulo que se quiere representar en coordenadas globales. Estos serán los nuevos valores para la variable *límite*. Es importante restar el desplazamiento de las coordenadas globales y no antes.

Después de actualizar la variable *límite* y calcular el nuevo *desplazamiento* y *entorno* se representa la nueva imagen.

En la figura 2.3.1.11 se muestra cómo el usuario selecciona el área sobre la que se aplicará el zoom o el windowing.

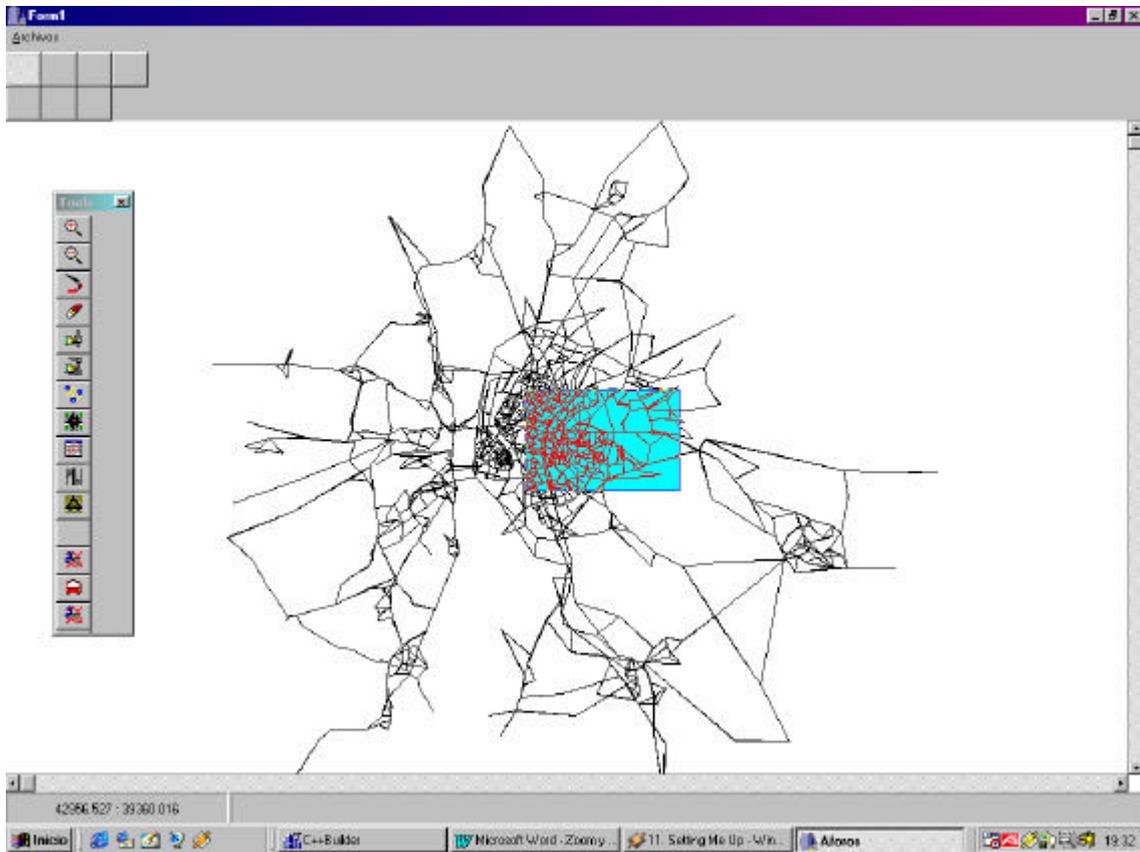


Figura 2.3.1.11: Selección del área sobre la que se aplica Zoom o Windowing

A continuación se presenta matemáticamente el procedimiento para realizar el zoom. Se denotarán por $(xz1, yz1)$ y $(xz2, yz2)$ los dos vértices del rectángulo que el usuario ha seleccionado como área a la que aplicar el zoom. En primer lugar se convierten las coordenadas de estos puntos a coordenadas globales:

$$x1 = f(xz1, yz1, entorno, limite)$$

$$x2 = f(xz2, yz2, entorno, limite)$$

A continuación debe corregirse el desplazamiento:

$$Nueva_x1 = x1 - desplazamiento_x$$

$$Nueva_y1 = y1 - desplazamiento_y$$

$$Nueva_x2 = x2 - desplazamiento_x$$

$$Nueva_y2 = y2 - desplazamiento_y$$

El rectángulo definido por $(Nueva_x1, Nueva_y1)$ y $(Nueva_x2, Nueva_y2)$ es el nuevo rectángulo *límite*. Por lo tanto, se actualiza la variable *límite* y se calcula de nuevo la variable *entorno*. Finalmente se centra la nueva imagen, es decir, se calcula *desplazamiento_x* y *desplazamiento_y* y se representa la nueva imagen.

El procedimiento para calcular el windowing es muy similar al de zoom:

- 1- De forma análoga al zoom se determina el área sobre la que se va a aplicar. Esto parece no tener mucho sentido, ya que tras realizar el windowing se representarán en pantalla igual o mayor número de nodos y tramos de los que se visualizan en ese instante. El matiz es que el centro del área seleccionada aparecerá tras el windowing en el centro de la imagen, y que el ampliar más o menos la porción de red a visualizar depende de las dimensiones del área seleccionada.
- 2- De nuevo se usan las variables *entorno* y *límite* para obtener las variables globales. Análogamente al caso del zoom, se debe restar a estos valores el desplazamiento posteriormente.
- 3- Se calculan dos cocientes, uno para cada lado, entre el rectángulo *límite* y la porción de área seleccionada. Estos cocientes reciben los nombres de *razon_x* y *razon_y*.
- 4- Se calcula el nuevo *límite* en coordenadas globales.

$$Nuevo_lx1 = lx1 - (x1 - lx1) * razon_x;$$

$$Nuevo_lx2 = lx2 + (lx2 - x2) * razon_x;$$

$$Nuevo_ly1 = ly1 - (y1 - ly1) * razon_y;$$

$$Nuevo_ly2 = ly2 + (ly2 - y2) * razon_y;$$

- 5- A continuación se actualiza la variable *límite* a sus nuevos valores y se recalcula la variable *entorno* y *desplazamiento* para representar la nueva imagen tras borrar la actual.

Matemáticamente, y usando la misma notación que para el caso del zoom:

$$x1 = f(xz1, yz1, entorno, limite)$$

$$x2 = f(xz2, yz2, entorno, limite)$$

A continuación debe corregirse el desplazamiento:

$$Nueva_x1 = x1 - desplazamiento_x$$

$$Nueva_y1 = y1 - desplazamiento_y$$

$$Nueva_x2 = x2 - desplazamiento_x$$

$$Nueva_y2 = y2 - desplazamiento_y$$

Se calculan $razon_x$ y $razon_y$:

$$razon_x = \frac{lx2 - lx1}{x2 - x1}$$

$$razon_y = \frac{ly2 - ly1}{y2 - y1}$$

Se calculan los nuevos valores para las coordenadas de los vértices del rectángulo límite:

$$lx1 = lx1 - (Nueva_x1 - lx1) \times razon_x$$

$$ly1 = ly1 - (Nueva_y1 - ly1) \times razon_y$$

$$lx2 = lx2 + (lx2 - Nueva_x2) \times razon_x$$

$$ly2 = ly2 + (ly2 - Nueva_y2) \times razon_y$$

Puede apreciarse como la nueva imagen a representar tendrá más resolución si el área seleccionada para realizar el windowing es más pequeña.