

# User-Centric Time-Distance Representation of Road Networks

Christian Kaiser<sup>1,2</sup>, Fergal Walsh<sup>1</sup>,  
Carson J. Q. Farmer<sup>1</sup>, and Alexei Pozdnoukhov<sup>1,\*</sup>

<sup>1</sup> National Centre for Geocomputation, National University of Ireland, Maynooth

<sup>2</sup> Institute of Geography, University of Lausanne, Switzerland

alexei.pozdnoukhov@nuim.ie

**Abstract.** This paper presents a new algorithm for computing time-distance transformations of a road network based on modified multi-dimensional scaling. The algorithm is designed to perform on a real-world road network, and provides alternative visualisations for travel time cognition and route planning. Several extensions are explored, including user-centric and route-centric road map transformations. Our implementation of the algorithm can be applied to any locality where travel time road network data is available. Here, it is illustrated on road network data for a rural region in Ireland. Limitations of the proposed algorithm are examined, and potential solutions are discussed.

**Keywords:** Road networks, time-distance cartograms, multi-dimensional scaling.

## 1 Introduction

With the increasing availability of navigation systems and interactive maps, spatial information is becoming increasingly accessible to the average user. Geographical maps have the potential to display large amounts of spatial information which can lead to ‘information overload’, unless an emphasis is placed on intelligent and adaptive presentation of information. Traditionally, to increase the amount of information contained within a map, additional contextual layers are required. The display of time information however, requires special consideration.

One alternative visualisation of spatial data comes from the use of time-distance mapping. In this context, a time-distance map is based on the travel time on a network (e.g. road network), where the space is deformed such that the length of the links (roads) between the nodes (intersections) of the network are proportional to their travel time. In this representation, the normal geographical scale (Euclidean distance) of the network can be replaced by a *time*-distance scale, using hours instead of kilometres as the unit of measurement. An interesting example of this type of representation is the time-distance transform of the London Tube Map with a static [1] and dynamic version [2]. These types of

---

\* Corresponding author.

artist-designed representations of time-distance currently appear to be the most graphically appealing maps; the average user may potentially find them more useful, as the deformation does not strongly interfere with the map content. The classic London Underground Map has been designed for readability and is already an approximation of a time-distance map, such that stations are at equal distances from each other, reflecting the fact that metro travel time is essentially independent of the real-world distance between stations [3].

Some work has been done in the past concerning automatic algorithms for displaying time-distance information [4,5,6]. While effective for geographic information professionals, these representations do not appear to be sufficiently intuitive to aid the average user in everyday decision making, as suggested by the fact that their use in maps for lay-people is uncommon. Conversely, cartograms [7,8,9] have been successful at representing areal data, giving promise to a similar representation of time-distance. To date, the attempts to integrate time with a geographical view of space either were aimed at professionals, or required considerable manual design intervention.

In this paper, we propose an automated algorithm designed to enhance the interpretation of road maps through the integration of the time dimension. Our algorithm is designed to maintain some characteristics of conventional geographical road maps, such as the orientation of the road segments and the location of selected landmarks. These constraints provide a time-distance representation which is sufficiently realistic, making it possible to be used as a complement or replacement for a traditional road map.

We propose two different implementations of the algorithm: (1) a user-centric mode in which the user is interested in features or locations which are in close proximity to their current location, and (2) a route-centric representation, where the user is interested in map features which are proximal to a specified route along the road network. Further, we provide a simple solution to the out-of-sample problem which provides a means of transforming satellite imagery, which may be used to provide context to the embedded road map visualisation. Our implementation of the algorithm is based on the OpenStreetMap (OSM) data ([www.osm.org](http://www.osm.org)) and can be applied anywhere where travel time data for road networks is available.

## 2 Previous Work

Time-space maps have had a long history in geography with works dating back to the 1960s [10,11]. Most time-distance maps are used to represent the distances between all nodes in a network as accurately as possible, and in order to achieve this, Multi-Dimensional Scaling (MDS) is commonly used [12,13,5]. MDS is designed to find the “best” configuration of network points by minimising a global error criteria [14]. However, MDS is not well suited for large networks with a high number of connections. In this case, the distortion becomes important and the visual representation can be quite confusing. Recently, Shimizu and Inoue [6] have created partial network time-space maps using a modified MDS approach,

where only a limited number of nodal distances are considered, for example the distances on the main transportation network. Additionally, they limit the deformation of the time-distance map by including the orientation of the network links in the constraints.

Denain and Langlois [15] have studied the utility of distance cartograms. In their opinion, isochrones are preferred because they are able to directly demonstrate time value-distances on the map without any geographical distortion. According to Denain and Langlois, the intent of making cartograms is to visualise a given phenomena, for example the reduction of the travel time due to the introduction of a high speed train network, without any link to geographic reality. But again, the authors focus on time-distance maps where all nodes on the network are involved.

User-centric representations of space have been successful in the past by integrating a local view into a global view, where the local view can be simply a zoomed view into the central region or be a somehow deformed map, e.g. a “fish-eye”. This type of method focuses on exploration of the map at two zoom levels: (1) a global view (context view) outside of the centre, and (2) a more detailed local view (focus view) in the centre of the map. For example, Yamamoto, Ozeki and Takahashi [16] explore a fisheye approach for dynamic web map services using a central region of focus surrounding the user’s location, along with a “glue region” connecting the two views. Similar studies combining a global and a local view have been conducted for network maps [17] or tourist maps [18].

We are not aware of any work that attempts to combine a topographic map together with a time-distance map with the purpose of enhancing the visual representation of a map from a user-centric perspective. In this paper, we present some possible approaches for transforming a traditional topographic or road map similar to those found in navigation devices and web-based maps, and provide a visualisation that may improve spatial cognition, and may also help to speed up decision making.

### 3 Methodology

MDS is a classic statistical technique which can be used to generate time-distance representations of spatial objects [14]. In Sec. 3.1, we review some of the relevant aspects of classic MDS in the context of time-distance mapping. In Sec. 3.2 to 3.6, modifications and extensions to classic MDS are introduced in order to improve the time-distance representation of real-world maps.

#### 3.1 Classic Metric Multi-dimensional Scaling

The task of classical MDS scaling is to find  $M$ -dimensional coordinates for a given set of objects  $\mathcal{V} = \{1, \dots, n\}$  such that their pair-wise Euclidean distances equal the given dissimilarities  $\mathbf{D} = \{d_{ij}\}$ . In other words, the problem to be solved is:

$$\begin{aligned}
&\text{Given } \mathbf{D} = \{d_{ij}\}, \quad (i, j) = 1, \dots, n, \\
&\text{find } x_1, \dots, x_n \in \mathbb{R}^M, \\
&\text{such that } \|x_i - x_j\| = d_{ij} \quad \text{for all } (i, j) = 1, \dots, n.
\end{aligned} \tag{1}$$

The solution is a configuration  $\mathbf{X} = \{x_1, \dots, x_n\}$  where each  $x_i$  is a point in the  $M$ -dimensional space corresponding to an object in  $\mathcal{V}$ . The complete explicitation to this problem, which has an exact solution if the dissimilarities  $d_{ij}$  are indeed Euclidean distances, was given by [19] using matrix algebra and eigenvalues analysis. Independently, an equivalent formulation some years later was described by [20]. An extension to the above method is possible if distances are known with errors, and is derived by minimization of a loss function operating with inner products. If one considers the dissimilarities as shortest path distances on a graph  $(\mathcal{V}, \mathcal{U})$  where  $\mathcal{V}$  are the vertices and  $\mathcal{U}$  the edges, the MDS is then a natural choice for graph drawing and network analysis [21], [22]. Classical scaling and its computational implementation is not, however, well suited for large scale problems, as it cannot handle missing values, or be adapted to favour a good fit to particular predefined pairs.

**Stress minimization.** As an alternative to classical scaling, one can consider the problem of directly minimising the misfit of the configuration to given dissimilarities [23]. This is done by finding a configuration  $\mathbf{X} = \{x_1, \dots, x_n\}$  such that we minimise the *stress* function

$$\sigma(\mathbf{X}) = \sum_{i < j} (d_{ij} - \|x_i - x_j\|)^2. \tag{2}$$

While no method is known for constructing algebraic optimal solutions that minimizes (2), this optimization problem can be solved iteratively with a form of gradient descent. To bring some flexibility to the setting, the influence of particular pairs of data points can be adjusted by introducing weights  $w_{ij} > 0$  as coefficients of the individual error terms:

$$\sigma_w(\mathbf{X}) = \sum_{i < j} w_{ij} (d_{ij} - \|x_i - x_j\|)^2. \tag{3}$$

If the desired distance is not known, unreliable, or an exact fit is not required for a pair, this pair can be simply ignored by setting  $w_{ij} = 0$ . Different scaling of the weights, of the general form  $w_{ij} = d_{ij}^{-p}$  can be considered, resulting for instance in Sammon's mapping [24] for  $p = 1$  and elastic mapping [25] for  $p = 2$ .

A "stress majorization" technique was developed [26] to improve the direct minimization of (3). The stress function is substituted with a *convex* function which is greater than or equal to the stress (hence majorization), but easier to minimize algebraically and computationally as it has a unique global minimum. In practice, a simple and efficient iterative procedure [27] can be used to find a configuration  $\mathbf{X}$ :

$$x_i^{[t+1]} \leftarrow \frac{\sum_{j \neq i} w_{ij} (x_j^{[t]} + s_{ij}^{[t]} (x_i^{[t]} - x_j^{[t]}))}{\sum_{j \neq i} w_{ij}}. \tag{4}$$

where

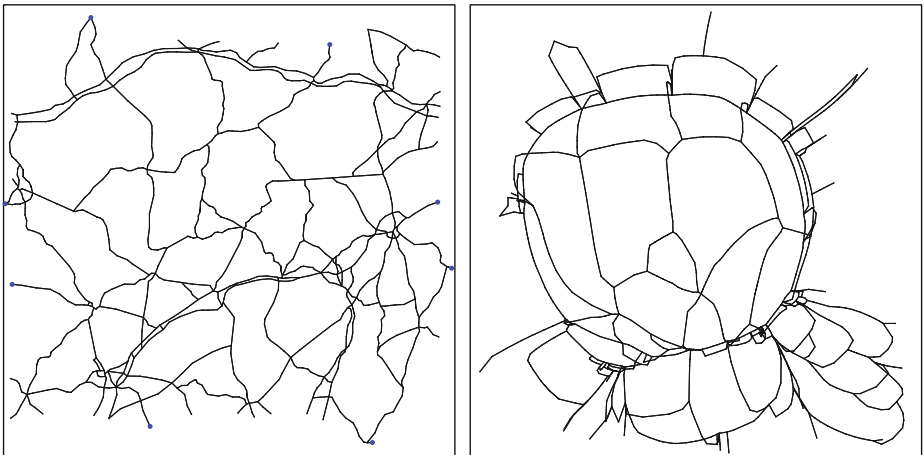
$$s_{ij}^{[t]} = \begin{cases} \frac{d_{ij}}{\|x_i^{[t]} - x_j^{[t]}\|} & \text{if } \|x_i^{[t]} - x_j^{[t]}\| < 0 \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

This process is repeated until the relative decrease in stress is larger than a predefined threshold value  $\epsilon > 0$ .

**Road network representations with MDS.** Consider a road network as a weighted undirected graph  $\mathcal{V}, \mathcal{U}$  of nodes with geographical coordinates  $\mathcal{V} = \{x_1^{geo}, \dots, x_n^{geo}\}$ . Let  $\mathbf{S}$  be a (sparse) matrix of distances between the adjacent nodes scaled according to the travel time. Define a shortest-path distance matrix on the graph computed from  $\mathbf{S}$  as  $\mathbf{G} = \{g_{ij}\}$ .

To apply MDS to a road network, one must compute the configuration  $\{x_1, \dots, x_n\}$  for  $\mathbf{D} = \mathbf{G}$  either by the classical method, or stress minimization. This produces a time-distance representation which attempts, on average, to reproduce all pairwise road distances. While these representations can be useful in particular applications, they suffer from a number of drawbacks. To motivate the following modifications to the MDS algorithms, consider an MDS applied to a particular road network, shown in Fig. 1. Note that in order to generate a more accurate representation of the travel times, additional nodes at regular intervals along the roads are added prior to the application of MDS. This also ensures that the most direct path in the time-distance map is the fastest one.

The global transformation shown in Fig. 1 (right) is difficult to interpret, due mostly to the fact that the transformation does not necessarily provide an intuitive representation of space. In other words, the map is overly warped, and distances and directions are difficult to interpret. Instead, we suggest a user-centric transform, which is aimed at (1) reproducing the travel distances



**Fig. 1.** Original road network (left) and its time-space representation obtained with classical MDS (right)

from a specific location (Sec. 3.2), (2) looking “similar” to a conventional map (Sec. 3.5), and (3) suited for graphical embedding in an untransformed map (Sec. 3.3). Using these extensions, we would like to produce navigation-oriented time-distance representations, which provides a desired time-distance alignment centred around a predefined path on a network (Sec. 3.4). Finally, we discuss the out-of-sample problem which has to be addressed to allow us to render other map geometries and warp satellite imagery of the area of interest (Sec. 3.6).

### 3.2 User-Centric Transforms

This extension allows the time-distance transformation to be centred around a given node  $x_c$  of the network. Let matrix  $\mathbf{E} = \{e_{ij}\}$  be a matrix of Euclidean distances between the nodes,  $e_{ij} = \|x_i^{geo} - x_j^{geo}\|$ . Our aim is to transform road distances in the vicinity of  $x_c$ , while maintaining the geographical layout of the distant nodes. We approach this by introducing a weighting scheme to favour for shortest path road distances in the vicinity of  $x_c$ . First, by using a smoothed Heaviside step function we define a local region of radius  $\rho > 0$  around  $x_c$ :

$$\lambda_{x_c} = \frac{1}{2}(1 + \tanh(\gamma(\rho\mathbf{I} - \mathbf{G}_c))), \quad (6)$$

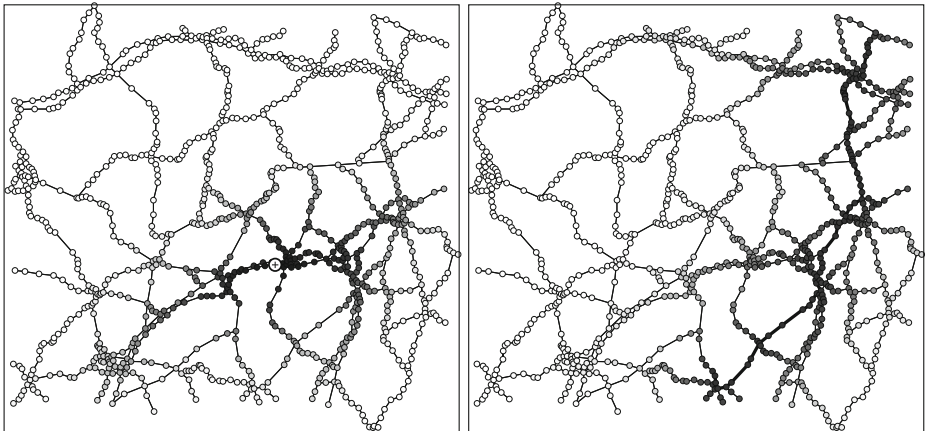
where steepness of the step is controlled by the parameter  $\gamma > 0$ ,  $\mathbf{I}$  is a  $(1 \times n)$  unit vector, and  $\mathbf{G}_c$  is a column of  $\mathbf{G}$  corresponding to  $x_c$ . The parameters  $\rho > 0$  and  $\gamma > 0$  allows one to adjust the geographical extent of the area to be transformed, and the smoothness of the transition from a non-transformed region into the time-distance representation of the vicinity of  $x_c$ . We then compute a weighted distance matrix  $\mathbf{D}_c$  with elements

$$d_{ij} = \frac{\lambda_{x_c}^i + \lambda_{x_c}^j}{2} g_{ij} + (1 - \frac{\lambda_{x_c}^i + \lambda_{x_c}^j}{2}) e_{ij}, \quad (7)$$

where the superscripts  $i$  and  $j$  denote vector elements. The effective distance matrix  $\mathbf{D}_c$  includes shortest path distances for all nodes in the vicinity  $\rho$  of  $x_c$  (shown with dark colours in Fig. 2 (left)) and preserves Euclidean distances between distant nodes (those in white colour in Fig. 2 (left)). It is then used for deriving a transformed network configuration with iterative optimization (4)-(5).

### 3.3 Preserving Landmark Locations

An exact match to pre-defined locations  $\{x_k\}$  can be required for a subset of nodes  $\{x_k^{geo}\}_{k=1,\dots,K}$ . It is particularly useful when one would like to keep the exact geographical location  $x_c^{geo}$  of the  $x_c$  and a set of boundary nodes in order to embed a transformed network into an untransformed road map. Enforcing the hard constraints  $x_k = x_k^{geo}$ ,  $k = 1, \dots, K$  would significantly complicate the optimization problem. However, a reasonable approximation can be achieved in a mean-square sense by adding a penalty  $\sum_{k=1}^K (x_k - x_k^{geo})^2$  with some weight  $\beta$  to the stress function. This approach falls under the “soft constraints” category



**Fig. 2.** Left: transformation vicinity for user-centric transform as defined by weights  $\lambda_{x_c}$ , Eq. (6). User location is marked with a crossed circle. Right: route-centric weight  $\lambda_r$ , Eq. (10). The selected route is shown in bold line. A white-to-black grey scale level corresponding to a  $[0, 1]$ -interval of the values of  $\lambda_{x_c}$ ,  $\lambda_r$  is used in both graphs.

in [28]. Consequently, the optimization procedure (4) has to include the following update steps at every iteration:

$$x_k^{[t+1]} \leftarrow x_k^{[t]} - 2\beta(x_k^{[t]} - x_k^{geo}), \quad k = 1, \dots, K. \quad (8)$$

An adaptive weighting scheme can be used to improve the fit of the distances between landmark nodes  $x_k$  by increasing the values of respective columns and rows of  $\mathbf{W}$ .

### 3.4 Route-Centric Transforms

Consider a predefined path a user may have chosen to follow on the road network. Suppose it consists of a sequence of adjacent nodes  $\{x_r\}_{r=1, \dots, R}$ . In this case, we would like to produce a transform which preserves the time-distances between nodes along the selected route, while maintaining the pair-wise Euclidean distances between nodes not proximal to the route. Firstly, we compute the shortest path distances from all the nodes of the network to the selected route:

$$g_i^{path} = \min_{r=1, \dots, R} g_{ir}, \quad i = 1, \dots, n. \quad (9)$$

We then introduce a weighting scheme for those nodes close to the route in analogy with Eq. 6:

$$\lambda_r = \left[ \frac{1}{2} (1 + \tanh(\gamma(\rho \mathbf{I} - g_i^{path}))) \right], \quad i = 1, \dots, n. \quad (10)$$

Here,  $\lambda_r$  is a vector containing weights for every node on the network describing its proximity to the selected route (refer to Fig. 2 for an illustration). The radius

of this region is controlled by user-defined parameter  $\rho > 0$  and the steepness of the transition by  $\gamma > 0$ . These parameters define the geographical extent of the area to be warped into time-distances. This weighting scheme is used to compute the matrix of effective distances similarly to Eq. (7) for iterative stress minimization.

The next modification to the MDS algorithm allows us to warp the route and present it as a straight line on the graph, maintaining the map-like properties described above. Suppose the nodes on the shortest route include  $\{x_r\}_{r=1,\dots,R}$ . This can be achieved by iterating the updates:

$$x_r^{[t+1]} \leftarrow x_r^{[t]} - 2\beta_{path}(x_r^{[t]} - x_r^{path}), \quad r = 1, \dots, R, \quad (11)$$

where  $x_r^{path}$  are the desired locations of the nodes on the path, and  $\beta_{path}$  the stress function weight of the nodes. In the following applications, we align these nodes along a straight-line path connecting the origin ( $o$ ) and destination ( $d$ ) of the route, precisely reproducing the travel time distances along said route.

### 3.5 Preserving Angles

The MDS-related transform extensions described above may lead to highly deformed transformations. To ensure that the layout of the network remains similar to the original road network, we introduce a new term into the stress function which is designed to preserve the orientations of the road segments<sup>1</sup>. Consider the set of all road segments  $\mathcal{U}$ , assuming their orientations are given by the set of angles  $\{\alpha_{ij}\}_{(ij)\in\mathcal{U}}$ . The double index  $(ij)$  denotes all pairs of adjacent nodes of the network. We would like to preserve the road segments orientations by minimising

$$\Psi = \sum_{(ij)\in\mathcal{U}} w_{ij}^{path} ((x_i - x_j) - d_{ij}[\cos \alpha_{ij} \quad \sin \alpha_{ij}]^T)^2. \quad (12)$$

This is achieved by including the following update steps into (4):

$$\begin{aligned} x_i^{[t+1]} &\leftarrow x_i^{[t]} - w_{ij}^{path} \beta_{ang} \nabla_{alpha}(x_i), \\ \nabla_{alpha}(x_i) &= \sum_{j\in\mathcal{N}(i)} (x_i - x_j) - d_{ij}[\cos \alpha_{ij} \quad \sin \alpha_{ij}]^T, \end{aligned} \quad (13)$$

where summation in gradient computation  $j \in \mathcal{N}(i)$  runs over the nodes  $j$  adjacent to  $i$  and  $\beta_{ang}$  is the weight of the penalty term  $\Psi$  in stress minimisation related to the angle of the road segments adjacent to the nodes along the path. We use the distance to the route weights  $w_{ij}^{path}$  computed as the elements of the matrix  $\mathbf{W} = \mathbf{I} - \lambda_r^T \lambda_r$  with  $\lambda_r$  defined by Eq. (10) to allow for more flexibility around the route while preserving the original Euclidean distances towards the borders of the area of interest.

<sup>1</sup> Ideally one would also like to preserve the topology of the network.

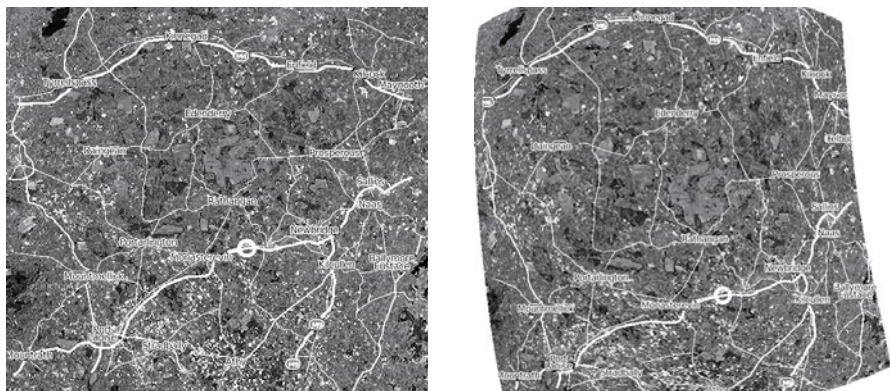


### 3.6 Out-of-Sample Problem

The transformation method as presented above provides a mapping of every node  $\mathcal{V}$  of the network from geographic space to the transformed space,  $F^{\text{MDS}} : \mathcal{V} \rightarrow \mathbb{R}^M$ . It does not however provide a mapping for any other locations in geographic space. This is due to the fact that in MDS, the embedded location of each sample (node) depends on it's distance from every other sample, such that all samples must be embedded at the same time. This is a limitation of MDS, known as the out-of-sample problem [29]. In order to map other features from geographical space (i.e. rivers, lakes, forests), or to transform a raster aerial photograph in the same manner as the road network, this problem must be solved (i.e. a mapping  $F : \mathbb{R}^N \rightarrow \mathbb{R}^M, \mathcal{V} \subset \mathbb{R}^N$  must be provided).

The solution proposed here takes advantage of the fact that the area bounded by a set of nodes in the transformed space represents the same area bounded by the same nodes in geographic space, with potentially different shape and size. This means that a continuous non-linear function which can reproduce the mapping of nodes from geographic to transformed space should be able to map any location from geographic space, as long as the location is within the bounds of the original network. Since it is not possible to derive such a transformation function from MDS, we can instead approximate the function using an artificial neural network (ANN) of multi-layered structure which is a powerful universal approximator for empirical dependencies [30]. This is done by starting with a discrete mapping of true geographical locations  $\{x_1^{\text{geo}}, \dots, x_n^{\text{geo}}\}$  to embedded locations derived from the MDS procedure  $\{x_1, \dots, x_n\}$ . The weights of the ANN are then iteratively adjusted until the ANN is able to closely approximate the discrete mapping, resulting in a continuous mapping function over the area of interest.

A raster image can then be transformed to produce a mapping of the geographic locations corresponding to each pixel in the original image to locations in the



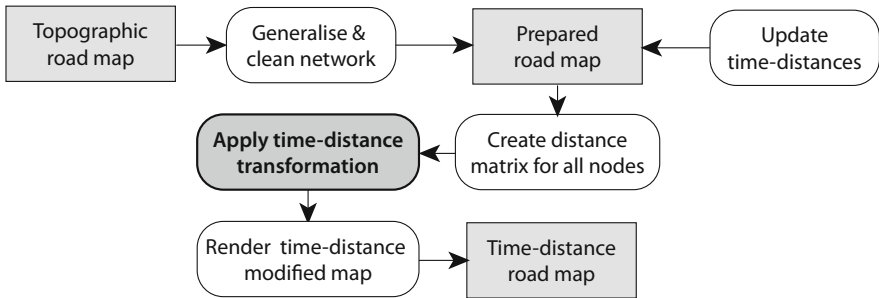
**Fig. 3.** Original satellite raster image (left) and its time-space representation using an ANN to approximate the modified MDS transform function(right). Note that the road network has been superimposed in both the original image and the transformed image to provide context.

transformed space using the above out-of-sample solution. Figure 3 shows this type of image transformation, and is potentially useful for navigation, as it includes features not found on the network itself, such as fields and other physical landscape features. Similarly, we are able to render vector-based map features.

## 4 Applications

We implemented a generalised framework for transforming geographical space based on data from the OpenStreetMap (OSM) project, within which the proposed time-distance transforms can be applied anywhere where suitable OSM data related to travel distance is available. Here, we have applied the methods covered in Sec. 3 to a series of examples based on OSM data for a region of roughly 50x50 kilometres west of Dublin, Ireland. The road network for the region includes 980 nodes (intersections), 1064 links (road-segments), and 5 different road categories. Figure 4 shows the general workflow for the time-distance map creation.

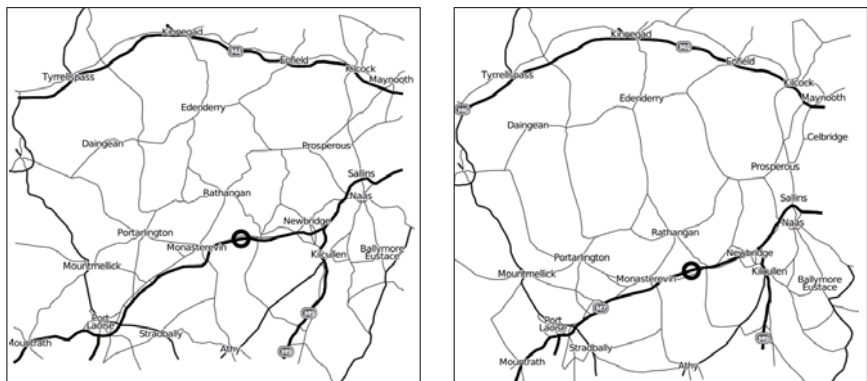
The most important stage is the introduction of time-distances for each link of the network. These distances can be provided dynamically from some other data sources as real time traffic services. For the current application however, travel times can be estimated from the network itself based on speed limits for the road categories. From this, we are able to generate a distance matrix between all points considered in the time-distance map, and feed it to the time-distance transformation, optionally followed by the out-of-sample extension. As a final step, the transformed map is rendered in a similar fashion to the original road network.



**Fig. 4.** Workflow for the creation of dynamic time-distance maps

### 4.1 User-Centric Time-Distance Map

Figure 5 (left) shows a conventional road map for the selected region, with a user-centric time-distance map (right), both rendered using the OSM style. Complete or partial topographic maps could also be represented in time-distance space, using the presented out-of-sample technique (Sec. 3.6). The time-distance representation shows the effect of the slower travel speed on the secondary roads (yellow) while motorways (depicted in blue) help reduce the time distances.

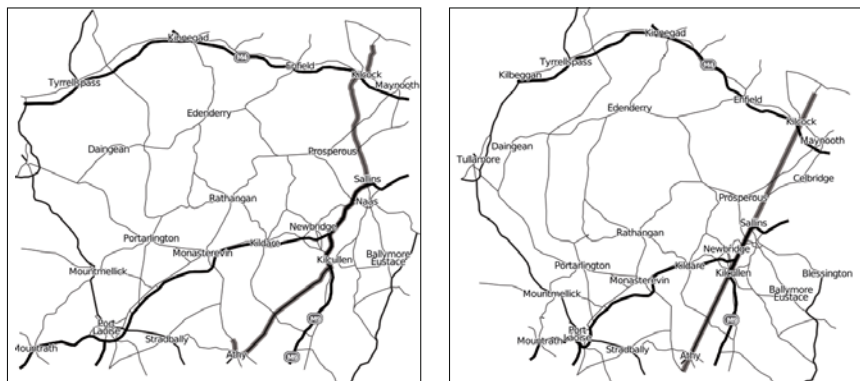


**Fig. 5.** User-centric traditional (left) and time-distance map created using OSM data

Note however, that the transformed map remains similar enough to the original road map to aid in interpretation. The map scale of the transformed map is based on time units around the position of the user. Places further away from the user's location are positioned somewhere between a topographic and time-distance representation to ensure an optimal interpretation and embedding into the topographic map space. However, in this case, border effects are present as no landmark nodes were defined. Normally, to embed the transformed space within a larger geographical space, one needs to define all outgoing routes at the border as landmarks, keeping the locations fixed as described in Sec. 3.3.

#### 4.2 Route-Centric Time-Distance Map

Figure 6 shows a selected route along a traditional (left) and time-distance map (right) computed as described in Sec. 3.4. In this route-centric map, the route

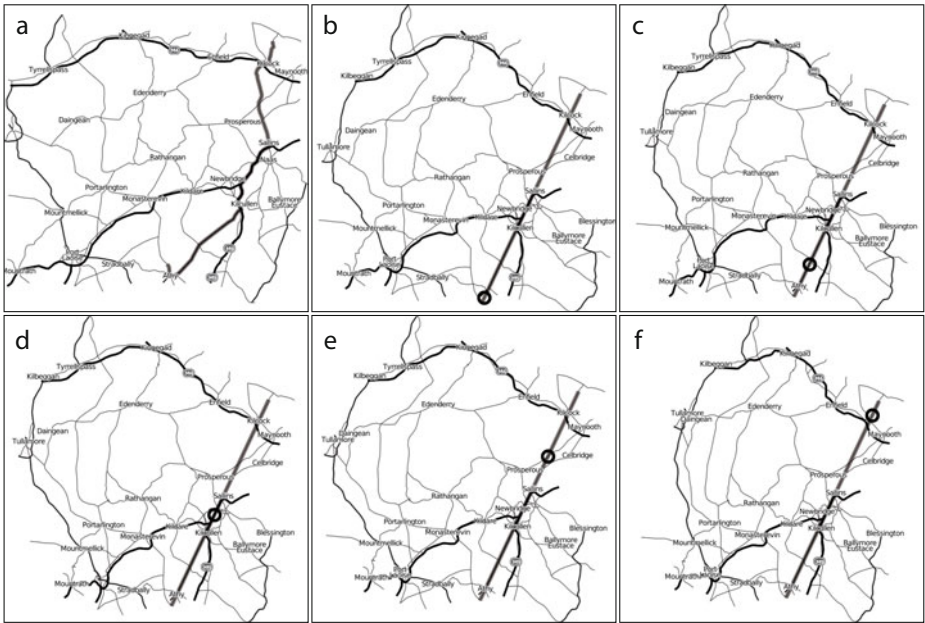


**Fig. 6.** Route-centric traditional (left) and time-distance map created using OSM data

becomes a straight line in the time-distance space. As such, objects surrounding the centred route are positioned according to their travel time from the route. This representation allows the mapping of additional objects along the road, such as tourist attractions, restaurants, or other facilities that may be of interest to the traveller. The user of the map has a fast, accurate view of the time needed to make a detour towards one of the surrounding features, or services. As such, the route-centric time-distance map can be a useful tool for planning trips. If desired, elements of topographic maps could also be included into such a map.

### 4.3 Dynamic Map Changes in the Route-Centric Representation

During a trip, the route-centric time-distance representation can potentially change constantly. If this is the case, it would be necessary to update the map dynamically, according to the current position of the user. For example, when driving down a motorway, the time-distances to the surrounding features will depend on the distance to the nearest exit. Conversely, if the user is close to the entrance to the motorway, all regions accessible via that motorway become closer. We have experimented with this idea by using a weighting scheme  $\mathbf{W} = \lambda_{x_c}^T \lambda_{x_c}$  in iterative stress majorization as location  $x_c$  changes while the imaginary user progresses along the route. Figure 7 shows several steps of a route-centric time-distance representation of a hypothetical drive along various type of roads in



**Fig. 7.** Route-centric traditional (a) and time-distance maps changing during a trip (b-f)

rural Ireland. The lengths of the road segments along the straightened route exactly match the driving time spent on each corresponding segment along the path. The time-distance map shows a number of changes throughout the drive, with objects and paths further away from the route remaining relatively stable.

## 5 Discussion and Conclusions

This paper discusses ways of enhancing the interpretation of road maps through integration of the time dimension in terms of network travel time. We have achieved this integration through the use of new algorithms for time-distance map transformation. In general, the algorithms presented here are designed to transform the geographical space of conventional road maps into a spatial-temporal hybrid map, where locations of network nodes are adjusted to reflect the time it takes to travel between them, while still maintaining some characteristics of the conventional geographical road map such as direction and orientation. More specifically, we provide several approaches for integrating time-distance maps in a user-oriented mapping environment using real world road maps. The first approach provides a user-centric visualisation in which map features are positioned relative to their travel time from the user. As such, features that are easily accessible in terms of travel time from the user's current position are placed closer to the user. The second approach focuses on representing the temporal proximity of features relative to a specified route of path along the road network. In this case, the visualisation is route-centric, whereby the main route is shown as a straight line path between point A and point B, and all other features are positioned by their time-distance from the nearest access point along the route. The geographical extent of the applied transform and the smoothness of the transition can be easily controlled with two parameters. The values of these have to be tuned to maximise the usability of the algorithm via cognition studies aimed to adjust its performance for different applications.

There remains many interesting avenues for future work on MDS-based time-distance transformations. The algorithm presented is not designed to strictly respect the topology of the network, which presents a potential limitation. While the tools to preserve topology are available, we have not yet explored them to the level required for proper implementation. Furthermore, improvements to our proxy out-of-sample extension are required, and a hybrid approach of piece-wise linear deformations based on triangulated irregular networks and conventional rubber-sheet warping are possible avenues for exploration.

The time-distance maps presented in this paper are potentially more useful in the context of in-car map navigation than classic space-time maps, due largely to the additional constraints designed to limit the level of network deformation. However, usability tests on decision making and cognition should be conducted in the future to provide additional insights into how useful such maps are for the average user. Such tests could also provide some indication as to how much a road network can be warped and still remain usable. Additionally, it should be studied which elements could be added to a time-distance map for facilitate the interpretation.

Issues remain when working with very large and detailed road networks where the cost of the required computations can become a problem for creating dynamic maps. Algorithmically we approached the task by developing an iterative procedure to minimise an MDS stress function which we compute with a customised pair-wise distance and weights matrix. While iterative nature of the stress majorization algorithm simplifies its adaptation for the use in a temporal setting, there are still convergence issues to be explored. Further work will show if our method is of reasonable efficiency for implementation as a server-side module delivering morphed maps to a mobile device working in a dynamic environment, where traffic conditions and user location and mode of transportation are constantly changing.

## Acknowledgements

Research presented in this paper was funded by a Strategic Research Cluster grant (07/SRC/I1168) and Stokes Programme by Science Foundation Ireland under the National Development Plan. The authors gratefully acknowledge this support.

## References

1. Karlin, O.: Time travel (2005) (Web page), <http://oskarlin.com/2005/11/29/time-travel:q>
2. Carden, T.: Travel time tube map, [http://www.tom-carden.co.uk/p5/tube\\_map\\_travel\\_times/applet/](http://www.tom-carden.co.uk/p5/tube_map_travel_times/applet/)
3. Böttger, J., Brandes, U., Deussen, O., Ziezold, H.: Map warping for the annotation of metro maps. *IEEE Computer Graphics and Applications* 28(5), 56–65 (2008)
4. Spiekermann, K., Wegener, M.: The shrinking continent: new time-space maps of Europe. *Environment and Planning B: Planning and Design* 21, 653–673 (1994)
5. Ahmed, N., Miller, H.: Time-space transformations of geographic space for exploring, analyzing and visualizing transportation systems. *Journal of Transport Geography* 15(1), 2–17 (2007)
6. Shimizu, E., Inoue, R.: A new algorithm for distance cartogram construction. *International Journal of Geographical Information Science* 23(11), 1453–1470 (2009)
7. Dorling, D.: *Area Cartograms: Their Use and Creation*. Geo Abstracts University of East Anglia, Norwich (1996)
8. Tobler, W.: Thirty five years of computer cartograms. *Annals of the Association of American Geographers* 94(1), 58–73 (2004)
9. Gastner, M.T., Newman, M.: Diffusion-based method for producing density equalizing maps. *Proceedings of the National Academy of Sciences of the United States of America* 101(20), 7499–7504 (2004)
10. Bunge, W.: *Theoretical geography*. PhD thesis, University of Washington (1960)
11. Tobler, W.: *Map transformation of geographic space*. PhD thesis, University of Washington (1961)
12. Marchand, B.: Deformation of a transportation space. *Annals of the Association of American Geographers* 63(4), 507–522 (1973)

13. Forer, P.: Space through time: a case study with NZ airlines. In: Cripps, E. (ed.) *Space-time concepts in urban and regional models*, Pion, London, pp. 22–45 (1974)
14. Kruskal, J.B., Wish, M.: *Multidimensional scaling*. In: *Quantitative applications in the social sciences*, Sage, Beverly Hills (1978)
15. Denain, J.C., Langlois, P.: *Cartographie en anamorphose*. *Mappemonde* 49(1), 16–19 (1998)
16. Yamamoto, D., Ozeki, S., Takahashi, N.: Focus+Glue+Context: an improved fish-eye approach for web map services. In: *17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Seattle, WA, November 4–6 (2009)
17. Sarkar, M., Brown, M.H.: Graphical fisheye views. *Communications of the ACM* 37(12), 73–84 (1994)
18. Guerra, F., Boutoura, C.: An electronic lens on digital tourist city-maps. In: *Mapping the 21st century: proceedings of the 20th International Cartographic Conference*, Beijing, pp. 1151–1157 (2001)
19. Torgerson, W.S.: *Multidimensional scaling: I. Theory and method*. *Psychometrika* 17, 401–419 (1952)
20. Gower, J.C.: Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika* 53(3–4), 325–338 (1966)
21. Kruskal, J.B., Seery, J.B.: Designing network diagrams. In: *Proceedings of the First General Conference on Social Graphics*, pp. 22–50 (1980)
22. Brandes, U., Pich, C.: Eigensolver methods for progressive multidimensional scaling of large data. In: Kaufmann, M., Wagner, D. (eds.) *GD 2006*. LNCS, vol. 4372, pp. 42–53. Springer, Heidelberg (2007)
23. Kruskal, J.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* 29(1), 1–27 (1964)
24. Sammon, J.W.: A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* 18(5), 401–409 (1969)
25. McGee, V.E.: The multidimensional scaling of elastic distances. *The British Journal of Mathematical and Statistical Psychology* 19, 181–196 (1966)
26. de Leeuw, J.: Applications of convex analysis to multidimensional scaling. In: Barra, J., Brodeau, F., Romier, G., Van Cutsem, B. (eds.) *Recent Developments in Statistics*, pp. 133–146. North Holland Publishing Company, Amsterdam (1977)
27. Gansner, E.R., Koren, Y., North, S.C.: Graph drawing by stress majorization. In: Pach, J. (ed.) *GD 2004*. LNCS, vol. 3383, pp. 239–250. Springer, Heidelberg (2005)
28. Borg, I., Groenen, P.J.F.: *Modern multidimensional scaling: theory and applications*. Springer, Heidelberg (2005)
29. Bengio, Y., Paiement, J.F., Vincent, P., Delalleau, O., Le Roux, N., Ouimet, M.: Out-of-sample extensions for lle, Isomap, MDS, eigenmaps, and spectral clustering. In: *Advances in NIPS*, pp. 177–184. MIT Press, Cambridge (2003)
30. Haykin, S.: *Neural networks: a comprehensive foundation*. Prentice-Hall, Englewood Cliffs (2008)