Fouille visuelle de données temporelles avec DataTube2

FATMA BOUALI

Université François Rabelais de Tours, Laboratoire d'Informatique, et Université de Lille 2

FREDERIC PLANTARD

Université François Rabelais de Tours, Laboratoire d'Informatique

AMINA BOUSEBA

Université François Rabelais de Tours, Laboratoire d'Informatique

GILLES VENTURINI

Université François Rabelais de Tours, Laboratoire d'Informatique

Résumé : Nous nous intéressons dans cet article à la fouille visuelle de données temporelles, où les données ont été mises sous la forme de n attributs dont les valeurs sont enregistrées pendant k instants. Après un état de l'art sur les différentes approches de visualisation de telles séries, nous présentons plus particulièrement une approche ayant reçue encore peu d'attention ("DataTube"). DataTube place les données dans un tube dont l'axe représente le temps. Nous étendons ensuite cette approche : tout d'abord nous définissons plusieurs modes de visualisations (couleurs, formes, etc) et nous ajoutons un axe temporel. Ensuite nous introduisons des interactions avec la possibilité de sélectionner des attributs et des instants, afficher des données complexes ou encore insérer des annotations sur la visualisation. Nous ajoutons une étape de classification non supervisée afin de regrouper dans la visualisation les attributs similaires. Enfin nous intégrons cette visualisation dans notre plateforme de fouille de données en réalité virtuelle VRMiner, avec un affichage stéréoscopique et des possibilités de navigation interactive. Nous appliquons cette visualisation sur plusieurs ensembles de données réelles et nous montrons qu'elle peut gérer jusqu'à 1,5 million de valeurs. Nous présentons également une évaluation utilisateur.

Mots clés : DataTube, Fouille de données, données temporelles, visualisation, 3D, réalité virtuelle, interactions, réorganisation.

Abstract: We deal in this paper with visual mining of temporal data, where data are represented by n time-dependent attributes (or series). We describe the state of the art in temporal data visualization, and we concentrate on a specific visualization (DataTube) which has received yet little attention. DataTube uses a tubular shape to represent the data. The axis of the tube represents the time. We perform several extensions to this visualization: we define several visualizations (color, shapes, etc) and we add a temporal axis. We introduce several interactions with the possibility to select attributes and time steps, or to add annotation on the visualization. We add a clustering algorithm in order to cluster together the attributes with similar behavior. We integrate this visualization in our data mining virtual reality platform VRMiner (with stereoscopic display and interactive navigation). We apply this visualization to several real-world data sets and we show that it can deal with 1,5 million values. We present also a user evaluation.

Key words: DataTube, Data Mining, temporal data, Visualization, 3D, Virtual Reality, Interactions, Rearrangement clustering.

1. INTRODUCTION

Nous nous intéressons dans ce travail au problème de la fouille visuelle et interactive de données temporelles. La dimension temps intervient dans de nombreux problèmes de fouille de données et elle a donné lieu à la définition de différents problèmes (analyse, prédiction, modélisation) et à l'étude de multiples méthodes [Antunes and Oliveira 2001]. Parmi ces méthodes, certaines font appel à des procédés d'extraction des connaissances entièrement automatiques, d'autres au contraire vont plutôt faire appel à des visualisations. Ces visualisations peuvent être utilisées en prétraitement pour permettre à l'expert de mieux comprendre les données, ou encore en post-traitement pour analyser visuellement les résultats d'une méthode de fouille, ou même de manière interactive pour découvrir des connaissances.

Parmi l'importante variété des données temporelles (séries numériques ou symboliques, suite d'événements, textes, images, sons, graphes, etc.), nous allons plus précisément considérer le cas de n attributs numériques prenant des valeurs sur k instants et décrivant, à chaque instant t considéré, l'évolution d'un phénomène donné. Les objectifs de l'expert que nous souhaitons prendre en compte sont par exemple : observer simultanément l'évolution de tous les attributs, détecter les valeurs manquantes, les événements particuliers et les périodicités éventuelles, les dépendances entre attributs et notamment les attributs se comportant de manière similaire, ou encore observer les conséquences d'un événement particulier situé dans le temps. En outre, nous souhaitons pouvoir afficher de grands volumes de données et permettre à l'expert d'annoter la visualisation pour partager les connaissances extraites. A titre d'exemple, les données traitées peuvent être des cours de la bourse (une variable représente un cours), de consommation (chaque variable décrit la consommation d'un produit) ou encore des données médicales (appareils de mesure).

Parmi les visualisations utilisées pour les données temporelles, il en existe une en particulier appelée « DataTube » [Ankerst 2000] qui a reçu peu d'attention et qui pourtant nous paraît très prometteuse. Comme nous allons le voir, DataTube est assez proche des méthodes orientées pixel pour la visualisation [Ankerst 2001] et qui associent une valeur à chaque pixel. Ces méthodes font partie de celles qui peuvent représenter de très grands volumes de données. Par ailleurs, DataTube est comme un "tube temporel" en 3D ce qui la pré-destine à être utilisée dans un environnement de réalité virtuelle permettant de percevoir la profondeur. Nous avons donc décidé d'étendre cette visualisation dans les directions suivantes :

- compléter les éléments visuels qui la composent,
- définir des interactions,
- utiliser des algorithmes de classification pour réorganiser les éléments visuels,
- intégrer cette visualisation dans un environnement de visualisation stéréoscopique et en particulier dans notre plateforme VRMiner [Azzag et al. 2005],
- tester la visualisation interactive obtenue sur des données réelles et avec des volumes beaucoup plus importants qu'auparavant.

La suite de cet article est organisée ainsi : dans la section 2 nous détaillons les approches de visualisation et de fouille visuelle de données temporelles, et plus particulièrement la visualisation DataTube. Dans la section 3 nous détaillons les extensions menant à la définition de DataTube2. Dans la section 4, nous présentons

les résultats expérimentaux sur des données réelles, et dans la section 5 nous détaillons une évaluation utilisateur. Enfin nous concluons et présentons des perspectives dans la section 6.

2. ETAT DE L'ART EN VISUALISATION DE DONNÉES TEMPORELLES ET PRINCIPES DE DATATUBE

Le domaine de la visualisation et de la fouille visuelle de données temporelles existe depuis longtemps [Minard 1861] (voir par exemple un survol dans [Muller and Schumann 2003]). Pour donner un aperçu de ce domaine, nous commençons donc par considérer le cas d'une séquence de symboles avec la visualisation "Arc Diagrams" [Wattenberg 2002]. Cette visualisation permet de détecter des motifs répétitifs en les faisant apparaître visuellement, mais elle ne peut traiter qu'une seule séquence de symboles ($n = 1$ dans notre notation). Une deuxième visualisation classique pour les données temporelles sont les spirales ([Carlis and Konstan 1998], [Weber et al. 2001]). Le centre de la spirale représente l'origine du temps. Ensuite le rayon de la spirale augmente et un même intervalle de temps T est toujours représenté par un seul tour de spirale. Ces méthodes peuvent visualiser des valeurs symboliques ([Weber et al. 2001] pour $n = 1$). Elles peuvent visualiser en 2D deux séries [Weber et al. 2001]. Dans [Carlis and Konstan 1998] plusieurs séries peuvent être visualisées en passant à une représentation 3D : jusqu'à $n = 12$ attributs visualisés sous forme d'histogrammes placés sur une spirale 2D, et jusqu'à $n = 112$ attributs en utilisant un empilement 3D de spirales 2D. Les spirales permettent de détecter des périodicités en ajustant interactivement l'intervalle T. Elles restent limitées à de petits volumes de données à la fois du point de vue du nombre de variables et du nombre d'instants considérés.

Dans les domaines où les échelles temporelles le permettent, la métaphore du calendrier ou de l'agenda fait partie des représentations classiques pour des événements ordonnés dans le temps. Ainsi nous pouvons citer par exemple [Daassi et al. 2000], mais également [van Wijk and van Selow 1999] où une étape de classification est effectuée pour rassembler les jours où la variable observée se comporte de manière identique, ou encore [Ankerst et al. 1996] où chaque événement fait l'objet d'un pixel d'une couleur donnée et où les jours du calendrier sont ensuite remplis par tous les événements ayant eu lieu ce jour là. Notons que la métaphore du crayon (l'axe du crayon représente le temps, et les facettes du crayon servent à visualiser l'évolution de variables) peut être utilisée notamment dans la visualisation de données sociologiques [Francis and Pritchard 2003]. Le nombre de variables reste limité pour ces visualisations.

Les plus grands volumes de données temporelles sont visualisés soit par des méthodes effectuant une étape de classification comme dans [Hébrail and Debregeas 1998] où 2665 courbes de 144 valeurs chacune sont regroupées en 100 classes avec une carte de Kohonen (chaque classe est visualisée sous la forme d'une courbe à 144 valeurs), soit dans les méthodes orientées pixels comme "Recursive Pattern" [Keim et al. 1995] où 530000 valeurs sont visualisées, ou encore "Circle segments" [Ankerst et al. 1996], ou bien encore "DataJewel" [Ankerst et al. 1996].

Nous nous sommes donc intéressés plus particulièrement aux travaux de ces auteurs. En ce qui concerne les grands volumes de données, citons la visualisation "Time Tube" [Chi et al. 1998] qui visualise spécifiquement les accès à un site Web avec une représentation arborescente où l'arbre représente les pages et l'épaisseur des arcs représente le nombre d'accès (voir section 4.2.1).

	t_1	t_2	...	t_k
A_1	$A_1(t_1)$	$A_1(t_2)$	...	$A_1(t_k)$
A_2	$A_2(t_1)$	$A_2(t_2)$	...	$A_2(t_k)$
...	...	...	...	...
A_n	$A_n(t_1)$	$A_n(t_2)$	...	$A_n(t_k)$

Fig. 1. Matrice des données

Enfin, comme mentionné dans l'introduction, notons qu'il existe bien d'autres types de données temporelles mettant en oeuvre des méthodes interactives et visuelles. Par exemple, on peut citer [Theron 2006] pour la visualisation de données temporelles hiérarchiques utilisant la métaphore "tree rings" (cernes du bois), ou encore la visualisation de graphes évoluant au cours du temps comme dans les travaux portant sur les réseaux sociaux [Bender-deMoll and McFarland 2006], ou encore la visualisation de la communication et coopération entre des personnes (voir un survol dans [Otjacques 2008]).

La visualisation "DataTube" [Ankerst 2001] peut être classée dans les méthodes orientées pixel même si elle s'en distingue par une différence importante provenant de l'utilisation de la 3D. Les valeurs de la matrice Attributs × Temps sont représentées par des codes de niveaux de gris, et les deux bords de cette matrice qui correspondent au temps sont repliés l'un sur l'autre pour former un tube. L'axe du tube représente donc l'axe temporel, et une "couronne" sur le tube représente la valeur des attributs pour l'instant considéré. "DataTube" a été appliquée à des données boursières ($n = 50$ cours). Ce type de représentation axiale du temps a été utilisé également dans "Kiviat tube" [Hackstadt and Malony 1994] où chaque instant de l'axe est un graphique de type "Star coordinates" [Kandogan 2000]. Cette méthode a été appliquée à la visualisation de la charge de $n = 64$ processeurs, mais nous ne l'avons pas retenue car elle peut générer des occlusions. Après avoir contacté Mihael Ankerst, nous avons appris que DataTube n'avait pas été plus développée : en particulier, elle n'a pas été testée au delà de $n = 50$ variables et pour des valeurs de $k > 100$. Pourtant, elle comporte des avantages et des potentiels comme nous allons le montrer dans la suite de l'article.

3. DATATUBE2

3.1 Définition de la visualisation

Nous notons par A_1, ..., A_n les n attributs numériques décrivant les données. Nous supposons que les valeurs de ces attributs ont été enregistrées initialement dans un intervalle de temps [0, T], de manière non synchrone. Ensuite nous pouvons définir des échelles temporelles différentes (heure, jour, mois, etc) et regrouper les instants dans des intervalles en effectuant la somme des valeurs des attributs dans cet intervalle. Notons que d'autres opérateurs "d'agrégation" seraient possibles comme par exemple la moyenne. Ces opérations de prétraitement sont incluses dans notre outil mais nous ne les détaillerons pas plus dans la suite : nous notons directement ces instants/intervalles de mesure par t_1, ..., t_k. Les données fournies à l'entrée de notre méthode de visualisation sont donc représentées par une matrice $n \times k$ et nous notons par $A_i(t_j)$ la valeur prise par l'attribut A_i à l'instant t_j (voir figure 1). Les

valeurs d'un attribut A_i sont ensuite normalisées sur un intervalle [0, 1] en considérant, à la demande de l'utilisateur, soit le minimum et le maximum de A_i, soit le minimum global et le maximum global, suivant la nature des données.

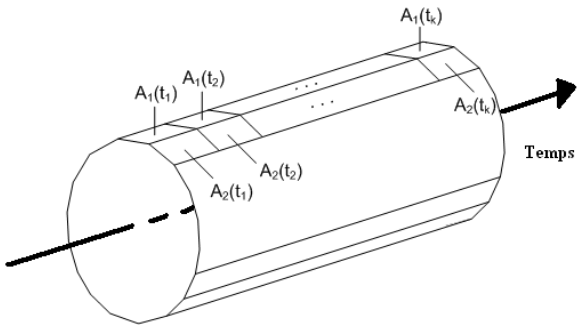


Fig. 2. Définition du tube temporel tel qu'initialement décrit dans DataTube

La visualisation de base dans DataTube (et DataTube2) consiste à représenter cette matrice sous la forme d'un tube temporel comme illustré sur la figure 2. Dans cette visualisation, chaque valeur A_i (t_j) est donc représentée par une facette rectangulaire. Un instant t_j est donc codé par un ensemble de facettes formant une couronne. L'évolution d'un attribut A_i est représentée par une ligne parallèle à l'axe du tube. Un niveau de gris est attribué à chaque facette en fonction de la valeur de A_i (t_j).

Les premières propriétés de cette visualisation sont les suivantes. Son mode de représentation est facilement compréhensible par des utilisateurs non spécialistes en fouille de données (d'après les présentations que nous avons pu en faire avec de vrais experts du domaine, voir section 4). Le tube peut être vu comme une métaphore perçue de manière immédiate par l'utilisateur. Tous les attributs sont visualisés simultanément ce qui permet de les comparer entre eux, mais aussi de détecter les valeurs manquantes. L'axe du tube représente le temps : la perception de l'écoulement du temps est donc très intuitive (nous verrons dans la section 3.5) que l'utilisation d'un écran stéréoscopique y contribue de manière significative). En naviguant à l'intérieur du tube, la perspective donne des effets visuels très intéressants : les données temporellement proches de la position de l'utilisateur dans l'espace 3D sont agrandies par rapport à celles qui se situent plus loin dans le temps et cet effet est accentué par la stéréoscopie dans DataTube2. Il en résulte donc un effet de "focalisation" (zoom) sur les instants proches et de conservation du contexte sur les instants suivants plus éloignés. L'utilisateur peut regarder le temps s'écouler et voir ce qui suit un événement, ou bien, en se retournant, voir ce qui précède un événement.

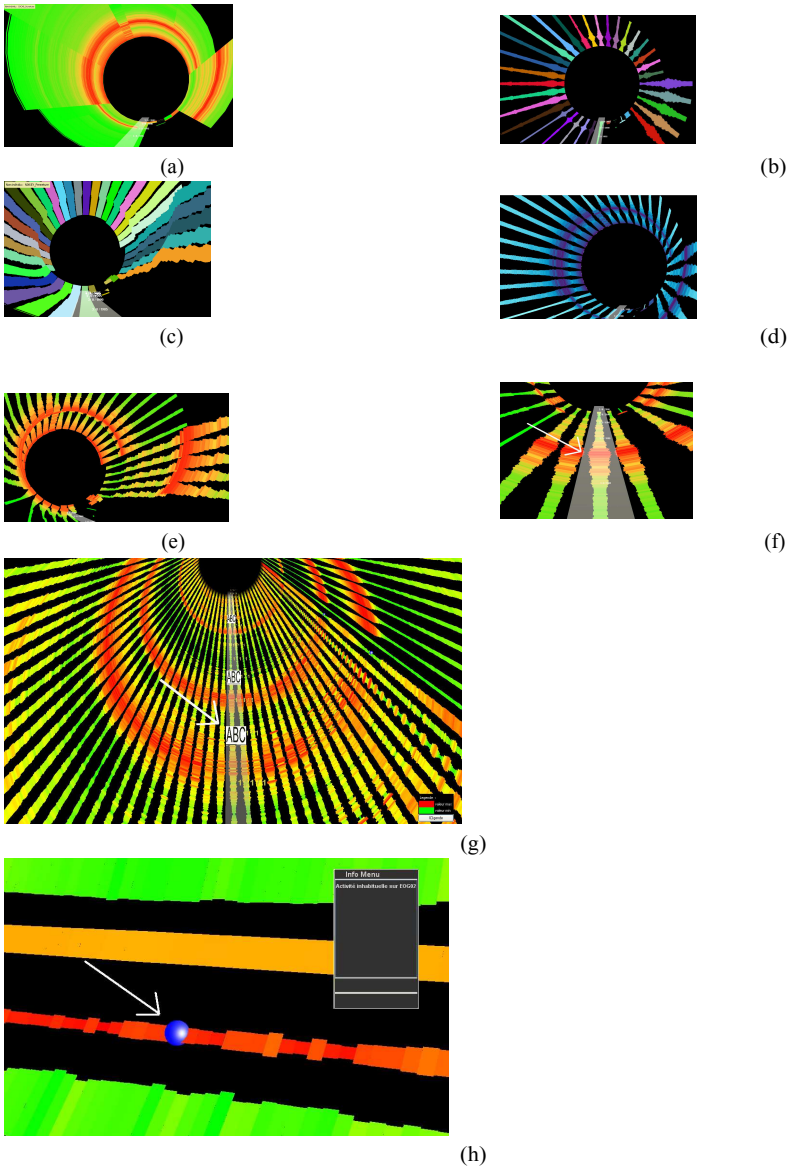


Fig. 3. Représentation des données par couleur (a), largeur (b), hauteur (c), en combinant couleur et largeur (d), couleur, largeur et hauteur (e). En (f) est représenté l'axe temporel. En (g) et (h), des exemples d'annotations ont été ajoutés dynamiquement

3.2 Extensions de la visualisation

Initialement, DataTube ne proposait qu'un seul mode de visualisation des données (facettes dont le niveau de gris dépend des valeurs de la variable correspondante). Nous avons donc proposé plusieurs modes de visualisation des valeurs $A_i(t_j)$

(voir figure 3) :

- par couleur : $A_i(t_j)$ est représentée par une face de taille constante dont seule la couleur va varier. Pour fixer la couleur, l'utilisateur peut choisir 3 couleurs qui correspondent aux valeurs minimum, médiane et maximum des données (par exemple Vert/Orange/Rouge). Le choix des couleurs peut dépendre du domaine traité,
- par largeur : la largeur des facettes représente les $A_i(t_j)$. L'utilisateur peut fixer l'amplitude maximale de cette largeur (par rapport à la place disponible sur le tube pour A_i),
- par hauteur : la hauteur des facettes varie en fonction de $A_i(t_j)$, ce qui peut donner des effets d'ondulation. Par contre, cette hauteur doit être faible sous peine de générer des occlusions, et l'utilisateur a la possibilité dans ce cas de contrôler l'opacité des facettes.

Ces modes de visualisation peuvent êtres combinés et permettent, suivant le type de données et le domaine traité, de faire apparaître des informations différentes, en particulier des combinaisons comme couleur et largeur (voir figure 3(d) et (e)), où le même attribut est représenté.

Les valeurs manquantes étant souvent présentes dans les bases de données réelles, nous considérons qu'elles représentent une absence d'événement dans l'intervalle de temps considéré. Nous avons donc fait le choix de les représenter par défaut dans une couleur donnée. Dans nos exemples, le fond de la visualisation étant noir, nous avons adopté cette couleur pour les valeurs manquantes dans toutes nos visualisations.

Nous donnons ensuite la possibilité à l'utilisateur d'ajouter un axe temporel pour représenter explicitement le temps, ce qui n'existait pas dans DataTube. Il n'y avait donc pas de moyen de se repérer temporellement dans la visualisation. Dans DataTube2, cet axe prend la forme d'un "chemin" composé de dalles, où chaque dalle représente un instant t_j (voir figure 3(f) et (g)). Ce chemin est placé initialement en bas et à l'intérieur du tube. Les dalles sont transparentes de manière à laisser percevoir les données situées en dessous d'elles. De plus, un label de type texte indique périodiquement à quel instant correspond une dalle. Ces dalles sont cliquables pour sélectionner un instant donné et le mettre en valeur par rapport aux autres. Enfin, l'utilisateur peut rajouter des annotations sur cet axe (voir section suivante).

3.3 Nouvelles interactions

En ce qui concerne la navigation dans la visualisation, l'utilisateur est initialement placé à l'extrémité de l'axe du tube et avec le regard vers l'intérieur du tube : il obtient ainsi une vue globale de toutes les données. Cette vue permet de voir par exemple les grandes tendances, les groupes de variables similaires (voir section 4) et les données manquantes. L'obtention d'un zoom se fait par effet de perspectives et par déplacement de l'utilisateur : les parois du tube sont proches de l'axe central, ce qui permet à l'utilisateur de les atteindre rapidement et d'observer localement les variables avec beaucoup de détails. Les déplacements ont lieu de manière classique (centrés sur l'utilisateur).

En ce qui concerne la sélection de données, chaque facette du tube est cliquable. Un clic gauche permet d'afficher en haut et à droite de l'écran le nom de l'attribut, l'instant considéré et la valeur $A_i(t_j)$. Un clic droit permet de mettre en avant (vers l'axe du tube) l'ensemble des facettes correspondant à A_i. Ainsi tout l'ensemble des valeurs d'un attribut peut être mis en avant pour le distinguer des autres et pouvoir l'observer en détail sans le confondre avec ses voisins.

L'utilisateur peut ajouter dynamiquement des annotations dans la visualisation. Ainsi il peut stocker des notes ou des repères relatifs aux informations et connaissances extraites, ou encore marquer un événement particulier afin de mieux en observer les causes ou conséquences. L'utilisateur peut aussi s'en servir pour faire une présentation interactive des résultats devant un ensemble de personnes. Comme le montre les figures 3(g) et (h), ces annotations prennent la forme d'un élément graphique comme une image choisie par l'utilisateur ou comme un repère visuel (sphère). Ces annotations peuvent être placées sur une facette du tube ou bien sur une dalle de l'axe temporel. On peut associer à ces annotations un lien (Web), un son ou encore un texte affiché dans une fenêtre (affichage contextuel).

3.4 Réorganisation des attributs similaires

L'ordre dans lequel sont présentés les variables autour du tube est important car il peut contribuer à faire apparaître des informations primordiales dans les données : il s'agit par exemple de découvrir des groupes de variables au comportement similaire, et de représenter de manière proche des groupes qui se ressemblent. Afin de résoudre ce problème nous avons utilisé un algorithme de classification qui réordonne la séquence de variables de manière à placer côte à côte les variables qui se ressemblent. Ce problème est bien connu en visualisation notamment dans le cas de la visualisation de matrices [Bertin 1977] [Jain et al. 1999] (voir [Climer and Zhang 2006] pour des travaux récents dans le domaine). Sa résolution implique la définition d'une mesure de similarité (ou de distance) entre les éléments à ré-ordonner, puis le choix d'un algorithme de réorganisation. Nous avons développé des approches heuristiques et génétiques pour ce problème notamment dans le cadre de visualisation de cubes OLAP [Sureau et al. 2009]. Dans ce travail sur DataTube2, nous donnons les résultats obtenus avec un algorithme standard dans ce domaine (BEA pour "'Bond Energy Algorithm"', [McCormick et al. 1972]). Cet algorithme utilise la mesure de similarité suivante entre deux variables :

$$ME(A_i, A_j) = \sum_{t=1}^{k} [A_i(t) * A_j(t)]$$

Elle est appelée "measure of effectiveness" et prend des valeurs maximales lorsque les deux variables sont identiques. BEA fonctionne de la manière suivante : il part d'une liste d'attributs notée L qui est initialement vide. Il commence par sélectionner deux attributs au hasard et les ajoute à L. Ensuite, il choisit parmi les attributs restants celui qui est le plus proche des attributs déjà dans L en considérant toutes les positions d'insertion possibles. Une fois que tous les attributs sont placés, le tube est réorganisé suivant l'ordre indiqué par L. Cette heuristique est déterministe et sa complexité est en $O(n^2)$.

Il est aussi possible de réorganiser les attributs en ne considérant qu'une partie des données dans le calcul de la similarité. Pour cela, l'utilisateur sélectionne un intervalle de temps donné. Ceci permet de ne pas tenir compte de certaines zones non intéressantes ou bruitées, mais surtout de vérifier si des comportements proches dans l'intervalle considéré le sont également à d'autres instants (corrélation).

3.5 Visualisation en stéréoscopie et intégration dans VRMiner

Comme nous l'avons mentionné précédemment, la perception de l'écoulement du temps dans DataTube2 est liée à l'axe du tube et à l'effet de perspective et de profondeur. Sur un écran 2D, notre visualisation ressemble à un disque et seuls les déplacements permettent d'avoir une idée de la forme tubulaire. Même si des effets d'opacité auraient permis de simuler la profondeur, nous avons opté pour une visualisation sur des écrans stéréoscopiques afin que la perception réelle de la 3D renforce la sensation de profondeur du tube et donc de l'écoulement du temps. Egalement, la navigation en 3D implémentée classiquement par une combinaison clavier/souris peut être améliorée pour l'utilisateur en utilisant des dispositifs adaptés. Nous avons donc intégré DataTube2 dans notre plateforme de fouille de données en réalité virtuelle VRMiner [Azzag et al. 2005]. VRMiner comporte initialement un grand écran stéréoscopique (écran polarisé) rétro-éclairé pour plusieurs utilisateurs et du matériel pour l'interaction (Flock of birds d'Ascension Technology Corporation, Wiimote de Nintendo, SpacePilot de 3DConnexion, gants de données d'Essential Reality). A ce jour (article en cours de préparation), VRminer peut fonctionner sous des environnements matériels plus légers dans le cadre d'une utilisation limitée à un utilisateur. Il s'agit par exemple d'utiliser un écran LCD 3D (modèle récent et bon marché type Samsung avec lunettes actives Nvidia). Dans ce poste mono-utilisateur, le SpacePilot est, en conjonction avec la souris, le périphérique de commande le plus approprié.

A la suite de cette implémentation, nous avons constaté que l'intégration de DataTube2 dans VRMiner permet d'améliorer la visualisation et d'augmenter les interactions et l'immersion dans les données. Ainsi, à l'aide de la visualisation stéréoscopique, on constate que la perception de la profondeur est importante en particulier pour bien appréhender l'écoulement du temps.

Il est intéressant de mentionner un travail précédent où des données temporelles de type log ont été visualisées en réalité virtuelle [Scullin et al. 1995]. La visualisation utilisée est un "scatter plot 3D" dans lequel de nombreux graphiques sont représentés (tous ceux que l'on peut obtenir en combinant par triplet jusqu'à $n = 10$ variables). Cependant, malgré l'utilisation de la transparence, cette visualisation engendre beaucoup d'occlusions. Du point de vue de l'application visée, les objectifs des auteurs étaient plus l'étude des performances des serveurs que l'analyse du comportement des utilisateurs. Ce système a été implémenté dans l'environnement CAVE [Symanzik et al. 1996] ainsi que dans des environnements plus légers utilisant des écrans CRT.

Bases	Nature des données	# valeurs	n	k
Polytech-Init	Log d'un site Web (documents et images)	1 447 839	9 463 éléments	153 jours
Polytech	Log d'un site Web (documents seuls)	563 668	1148 pages	491 jours
Antsearch	Log d'un serveur Web (plusieurs sites Web)	66 875	107 pages	625 jours
EEG	Electro-encéphalogrammes	67 136	64 électrodes	1049 × 2 ms
CONSO	Consommation d'une denrée (données confidentielles)	365 000	1000 individus	365 jours
BIOMED	Mesures d'un appareil à l'hôpital (glycémie)	30000	600 patients	selon le patient

Table I. Bases de données réelles testées avec les paramètres de visualisation (n, k).

4. RESULTATS SUR DES BASES REELLES

4.1 Données réelles étudiées

Les premiers tests de DataTube2 que nous ne décrivons pas ici ont porté sur des données réelles classiques telles que des cours de la bourse similaires à celles utilisées dans le seul test publié de DataTube initial, ainsi que sur des séries de l'INSEE.

Ensuite, nous avons appliqué notre visualisation aux différentes bases de données réelles présentées dans la table I avec, pour toutes les bases (sauf CONSO), un expert du domaine à qui nous avons montré les résultats obtenus. Comme nous le mentionnons dans la conclusion, nous sommes également en train d'implémenter DataTube2 chez un industriel dans le cadre de l'étude de l'évolution temporelle d'un système complexe. Les caractéristiques de ces données sont présentées dans la table I, et certaines d'entre elles faisant l'objet d'un accord de confidentialité, nous ne pouvons pas donner autant de détails que nous le souhaiterions.

4.2 Visualisation des accès à un site Web

4.2.1 *Approches connexes*. La fouille de données d'usage du Web est un bon champ d'expérimentation pour les approches visuelles et interactives car les données sont de nature hétérogène, en volume important, et tout le processus d'extraction de connaissances est tourné vers l'utilisateur final, qui n'est pas nécessairement un informaticien ou un expert en fouille de données. Il est donc important de proposer dans ce cadre des visualisations faciles à apprendre et permettant aussi une présentation à d'autres personnes. Un des premiers travaux dans le domaine concerne le système Webviz [Pitkow and Bharat 1994] qui affiche un graphe de pages Web où les liens entre les pages sont colorés en fonction des visites qu'elles ont reçues. Ce type de représentation a été utilisé dans bien d'autres travaux comme par exemple dans le système VisVIP [Cugini and Scholtz 1999] qui complète le graphe de pages par une courbe reliant les pages et symbolisant la navigation d'un internaute, ainsi que par des colonnes en chaque noeud symbolisant le temps passé sur chaque page. MIR [Kizhakke 2000] est l'une des rares représentations utilisant une métaphore : le site Web est une ville dont chaque bâtiment représente une page, et la navigation d'un utilisateur est

représentée par les déplacements d'un avatar dans la ville. Notons également qu'il existe des visualisations dynamiques de log qui présentent les données comme une séquence de représentations [Minar and Donath 1999] [Skog and Holmquist 2000], charge alors à l'utilisateur de mémoriser les instants précédents (raison principale pour laquelle nous n'avons pas retenu ce type d'approche).

Parmi les méthodes qui peuvent traiter les plus grands volumes de données, nous mentionnons à nouveau TimeTube [Chi et al. 1998] où les accès à 7588 pages Web sont représentés sous une forme arborescente. Ce système peut donc représenter la structure du site Web mais le temps y joue un mineur : seulement quelques instants différents sont visualisés. DataJewel [Ankerst et al. 1996] utilise une représentation de type calendrier ou chaque jour est rempli par des pixels représentant les événements (accès à telle page, utilisation de tel navigateur). Ce système utilise une métaphore aisément compréhensible (calendrier) mais cependant il ne permet pas de représenter facilement l'absence d'événements (aucune visite sur une page) ou encore de comparer entre eux les accès aux pages. Enfin, nous mentionnons une utilisation des cartes de Kohonen [Benabdeslem et al. 2002]) où les pages sont regroupées entre elles selon leur co-occurrence dans la navigation des utilisateurs. Notre approche se situe dans cette catégorie de méthodes où la priorité est donnée au traitement de grands volumes de données et à la découverte de connaissances avec un algorithme de classification.

Pour finir, il est important de mentionner les approches commerciales s'appuyant sur des représentations classiques comme par exemple Google Analytics. Ces outils sont très efficaces pour représenter l'activité d'une page, l'activité globale d'un site, l'origine des utilisateurs, etc. Cependant, si le site comporte beaucoup de pages, il est impossible de représenter leur activité simultanément (comme le fait TimeTube par exemple). De plus, l'orientation "grand public" de ces outils fait que, si l'on veut détecter des groupes de pages aux activités similaires, cela est impossible car le temps d'exécution d'un algorithme de classification côté serveur et à grande échelle est rédhibitoire.

4.2.2 *Données traitées*. Nous avons appliqué DataTube2 à différents logs obtenus à partir de sites Web réels (voir table I). La base "Polytech" contient un total de 1148 pages du site de Polytech'Tours (www.polytech-tours.fr). Le log a été traité de manière à extraire seulement les pages correspondant à des documents (pages html, php et asp). Les impacts sur ces pages ont été mesurés sur une période de 491 jours (plus de 500000 impacts). La base "Polytech-init" concerne le même site sur une période antérieure sans qu'aucun prétraitement n'ait été appliqué, ce qui ajoute principalement toutes les images (qui font chacune l'objet d'une requête lors du chargement d'une page Web). Ce log va être utilisé pour tester les limites de notre méthode. Enfin, la base "Antsearch" représente les accès aux différents sites hébergés par le serveur www.antsearch.univ-tours.fr. Tous les tests mentionnés dans la suite ont été réalisés sur un ordinateur standard (MacBook Pro, Intel Core Duo à 2.4GHz, RAM de 4Go).

Bases	Temps visualisation	Temps classification	Efficacité classification
Polytech	1.4 s	22.5 s	1.19
Polytech-Init	34 s	8 min.	1.03
Antsearch	407 ms	234 ms	1.39

Table II. Temps d'exécution (visualisation, classification) et efficacité relative de la classification (ratio après/avant de la somme des similarités entre variables successives).

4.2.3 *Evaluation quantitative*. Dans un premier temps, nous avons évalué quantitativement les performances de DataTube2 en ce qui concerne les temps d'exécution et l'efficacité de la réorganisation. Ainsi nous avons représenté dans la table II les temps d'exécution nécessaires pour la visualisation et pour la classification des données avec BEA. On constate que le temps de construction de la visualisation est tout à fait correct (implémentation en Java/Java3D). BEA ayant une complexité quadratique, on remarque que pour la plus grande base (Polytech-init) le temps nécessaire est de l'ordre de 8 minutes. Même si ce temps est beaucoup plus long que celui nécessaire à la construction de la visualisation, il reste cependant acceptable pour un tel volume de données. Dans cette même table II nous avons évalué quantitativement la performance de BEA en terme de classification. Pour cela, nous mesurons la qualité d'une classification des variables en faisant la somme des similarités entre variables successives prises 2 à 2. Plus cette somme est élevée, plus les variables voisines dans la représentation se ressemblent en termes de comportement au cours du temps. Initialement, les pages sont ordonnées par la date de leur première visite (ce qui correspond dans notre cas à la date de création). On constate que BEA améliore cette disposition initiale de manière significative. Cela se confirme également dans le cadre de l'évaluation utilisateur (voir section 5). Pour la base Polytech-init, le ratio mesuré n'est pas très important car les données sont très éparses (la similarité initiale entre les pages est très élevée car de nombreuses valeurs sont à 0).

Nous avons ensuite utilisé la base Polytech-init pour tester jusqu'à quel volume de données peut aller l'implémentation actuelle de DataTube2. Notons tout d'abord que pour toutes les autres bases testées, les interactions ont été parfaitement fluides pour l'utilisateur (nous n'avons pas mesuré précisément le nombre d'affichages par seconde mais à l'utilisation aucun problème de cet ordre ne s'est posé pour ces bases). Pour visualiser Polytech-init, les pages ont été ordonnées par leur date de création. L'aspect global de la visualisation ressemble donc à une sorte de spirale. Avec ce volume de données (environ 1.5 million de valeurs affichées), les interactions telles que les déplacements sont limités du fait de la lenteur de l'affichage (de l'ordre d'un affichage par seconde). Cette représentation doit donc être plutôt considérée comme statique même si on peut obtenir plusieurs points de vue sur les données. Nous espérons dans le futur pouvoir améliorer l'implémentation de DataTube2 afin de dépasser cette limite d'affichage.

4.2.4 *Découverte de connaissances et retour de l'expert*. Nous présentons ici des résultats pouvant typiquement être extraits avec DataTube2. Dans toutes les visualisations, nous avons représenté visuellement le nombre d'impacts par jour avec une couleur allant du vert (peu d'impacts) au rouge (grand nombre d'impacts).

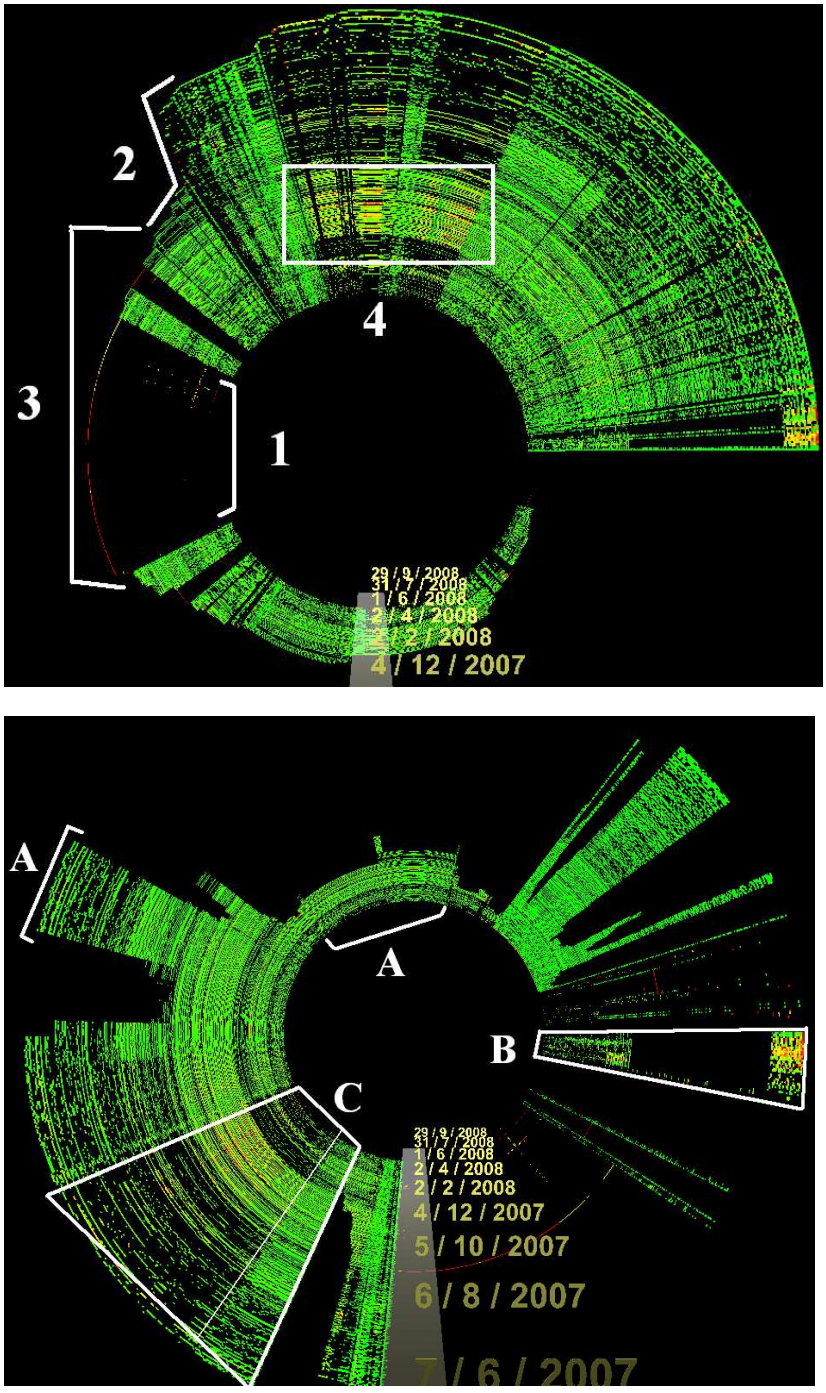
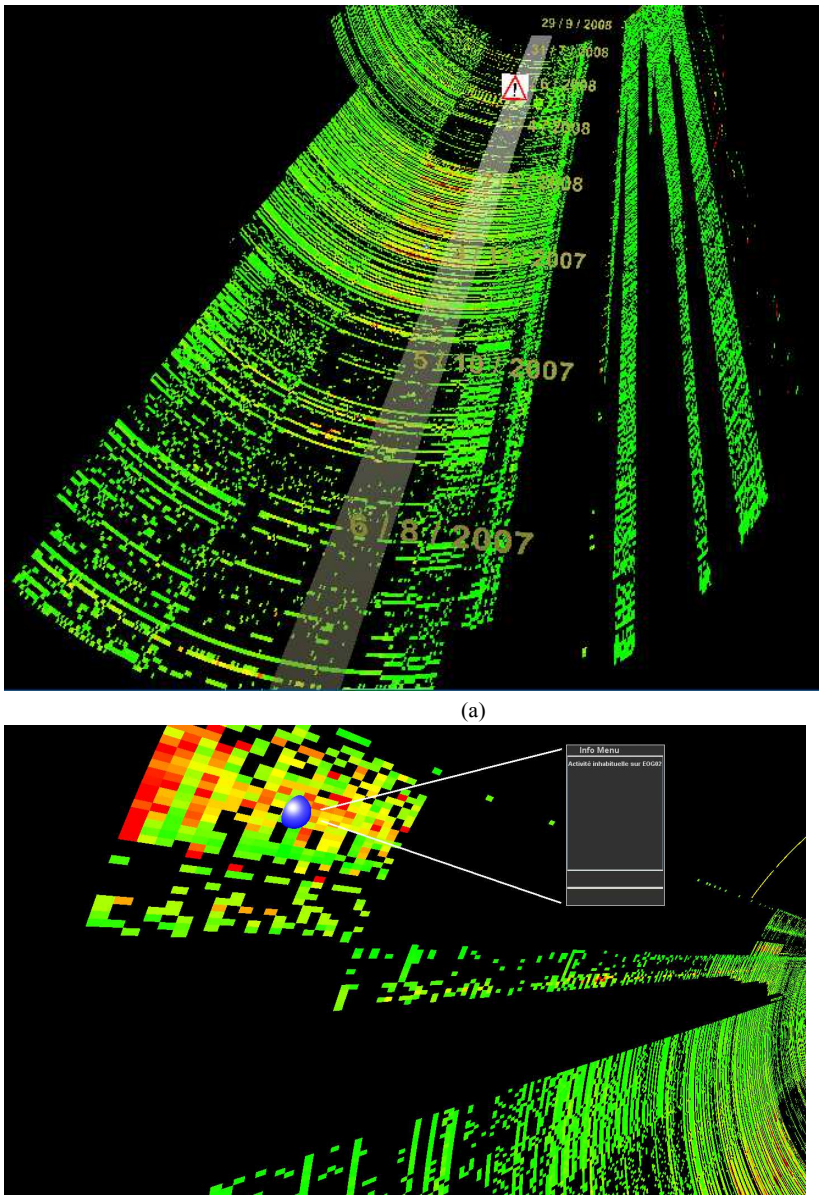


Fig. 4. Illustration de l'algorithme de classification sur la base Polytech : la première visualisation (haut) représente les pages triées par leur date de création, alors que la deuxième représente les pages à l'aide de l'algorithme de classification (voir section 4.2.4 pour plus d'explications).



(a)

(b)

Fig. 5. En (a), zoom sur la base Polytech, et en (b) un exemple de niveau de détail élevé.

Tout d'abord, l'utilisateur obtient une vue globale de toutes ses données comme sur la figure 4. En haut de cette figure, les pages sont ordonnées par leur date de création, ce qui donne cet effet de spirale. Il est très facile de détecter les pages qui, après leur création, n'ont plus fait l'objet de beaucoup de visites (voir la zone

marquée "1" sur la figure). Il est possible également de détecter les périodes où aucune page n'a été ajoutée au site (voir la zone "2"), ou au contraire les périodes où de très nombreuses pages ont été ajoutées (voir zone "3"). Dans cette figure, la zone "4" correspond à des pages qui, pendant un intervalle de temps donné, ont reçu plus de visites que les autres.

Dans l'image du bas de la figure 4, l'algorithme de réorganisation a été utilisé pour placer côte à côte les pages à l'activité similaire. On peut ainsi noter les différences entre l'image du haut (pages ordonnées selon la date de création) et celle du bas. L'algorithme de classification a tendance à regrouper les pages 1) créées à la même période, et 2) ayant ensuite une activité similaire. De nombreux groupes peuvent ainsi être identifiés, avec par exemple les deux groupes notés "A" sur la figure. Ensuite, des groupes plus spécifiques peuvent être identifiés. Les groupes "B" correspondent à des pages qui, après une période initiale de forte activité, n'ont reçu aucune visite, pour être de nouveau visitées ensuite. Le Webmestre peut ainsi détecter par exemple les pages temporairement inaccessibles. Enfin, le groupe de pages "C" est intéressant : il peut être subdivisé en deux sous groupes très homogènes que BEA a placé côte à côte dans la visualisation.

A partir de la vue globale, l'utilisateur peut obtenir des détails sur les données de log, aussi bien à un niveau intermédiaire qu'élevé. Par exemple, la figure 5(a) illustre un niveau de détail intermédiaire obtenu lorsque l'utilisateur se rapproche d'une partie de la visualisation. Dans la figure 5(b) nous illustrons un niveau de détail plus élevé encore pour un groupe de pages et un intervalle de temps donnés. Les données deviennent alors localement très précises. Même si en pratique cela ne semble pas avoir gêné les expérimentateurs, il faut noter que le contexte est cependant perdu en partie et que la localisation de la zone détaillée par rapport à l'ensemble fait appel à la mémorisation des déplacements qu'a pu faire l'utilisateur. Dans les perspectives de ce travail, nous comptons étudier comment remédier à cela si besoin, peut être en ajoutant une représentation miniature du tube.

Les Webmestres des sites concernés ont pu également tester DataTube2 avec leurs données de log, et nous avons pu ainsi recueillir leurs remarques. Ils ne connaissaient pas cette visualisation : nous avons donc constaté qu'une quinzaine de minutes ont suffi pour expliquer la structure de la visualisation (forme tubulaire, codage couleur des variables, axe temporel), ensuite les principales interactions (déplacements, sélections) et enfin ce qu'effectue l'algorithme de réorganisation. Ils ont été capables de détecter les informations mentionnées précédemment. Ils ont été particulièrement intéressés par la visualisation globale des données. Ils ont pu reconnaître certains groupes de pages (i.e. "Actualités", "Presse", "Galerie", "Cours", etc) d'après le regroupement effectué et la mise en lumière de comportements similaires. Pour la base Polytech, le Webmestre a étudié plus particulièrement l'influence de la période de fin des études/vacances/reprise des études sur les visites du site. Enfin, l'utilisation de l'environnement de réalité virtuelle a été bien perçue par les utilisateurs, notamment en ce qui concerne la présentation des données (voir section 5).

Enfin, par rapport aux deux visualisations mentionnées dans l'état de l'art et qui sont les seules à notre connaissance à pouvoir visualiser de grands volumes de

données de log, notons que Datatube2 visualise beaucoup plus de pages que DataJewel [Ankerst et al. 1996] (limité à quelques pages) et au moins autant de pages et beaucoup plus d'instants que TimeTube [Chi et al. 1998] (7500 pages, limité à quelques instants). En effet, nous avons également visualisé 9463 pages sur 153 jours en utilisant une échelle heure par heure, ce qui représente près de 1 500 000 valeurs.

4.3 Exploration d'électro-encéphalogrammes

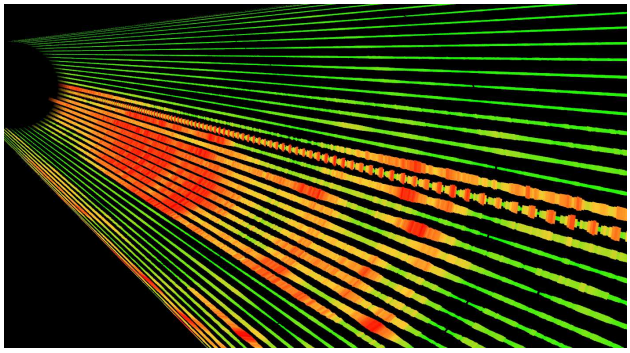
L'équipe Vieillissement et Mémoire du laboratoire Langage, Mémoire et Développement Cognitif (UMR 6215, Université de Poitiers et Université de Tours) nous a fourni des données issues d'électro-encéphalogrammes illustrant par des courbes la réaction de 64 zones du cerveau à des stimuli visuels [Fay et al. 2005]. Les données représentent donc l'évolution des 64 électrodes dans le temps, ici sur une durée de 2 secondes par pas de 2 ms. Notre objectif est d'évaluer l'intérêt de DataTube2 pour ce type de données et notamment la performance de l'algorithme de regroupement décrit précédemment. Ce regroupement est une information importante pour les experts du domaine traité.

Sur la figure 6(a) nous pouvons voir directement l'évolution de chaque électrode par rapport aux autres, les comparer grâce au regroupement effectué et évaluer les zones du cerveau actives lors de la réception d'un stimulus (ici l'apparition d'une image devant les yeux), ainsi que les répercutions sur les autres électrodes. La visualisation DataTube2 permet de voir directement les relations entre les électrodes et les signaux du cerveau, contrairement aux courbes couramment utilisées, visibles sur la figure 6(b). La figure 6(c) montre l'état des électrodes lorsque le patient cligne de l'œil et donc les zones actives du cerveau.

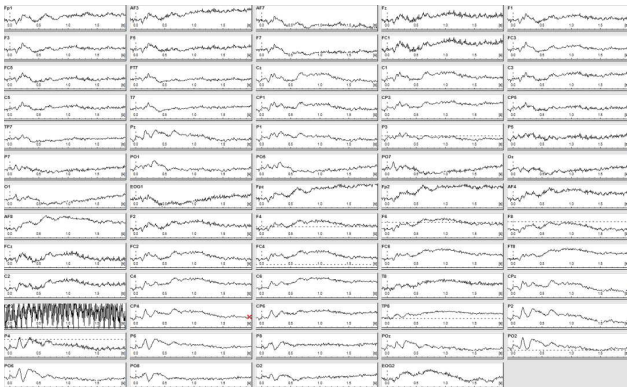
Les données fournies sont des données de tests et nous pourrons par la suite tester cette visualisation et la faire valider par les utilisateurs concernés avec des données complètes. Cependant lors de la présentation de la visualisation d'EEG à ces utilisateurs nous avons pu constater, comme pour les informaticiens précédents, que la prise en main de la méthode est relativement facile et rapide. Le plus appréciable dans cette méthode est de pouvoir visualiser et comparer les activités de toutes les électrodes très facilement contrairement au recueil de courbes qu'ils utilisent classiquement (figure 6 (b)).

4.4 Autres données testées

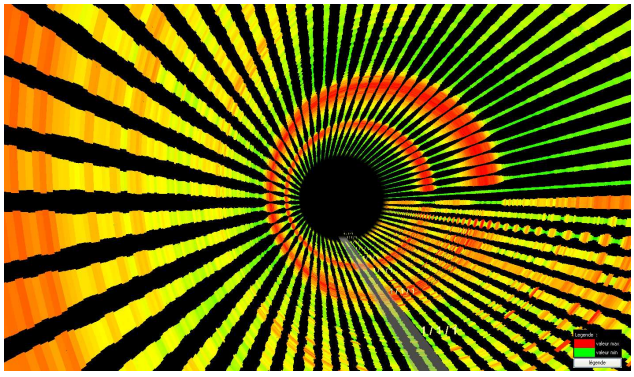
Nous avons appliqué DataTube2 sur les autres bases de données mentionnées dans la table I. Pour la base CONSO, il s'agit de représenter la consommation quotidienne d'une denrée pendant un an par 1000 personnes différentes. DataTube2 a permis de découvrir des comportements liés aux jours de la semaine mais également aux périodes de vacances comme par exemple celles liées aux différentes zones A, B ou C. Pour la base BIOMED (voir figure 7), il s'agit de représenter les mesures faites de la glycémie sur un ensemble de 600 patients (collaboration avec le Dr. Pierre Kalfon, Service de Réanimation au Centre Hospitalier de Chartres, et la société LK2). L'objectif de la visualisation est de fournir un aperçu initial des données, afin de contribuer à un projet plus vaste, le contrôle automatique de l'injection d'insuline chez des patients en réanimation



(a)



(b)



(c)

Fig. 6. Electro-encéphalogrammes obtenus lors d'un stimulus visuel en (a), application d'origine en (b), effet d'un clignement de l'oeil en (c)

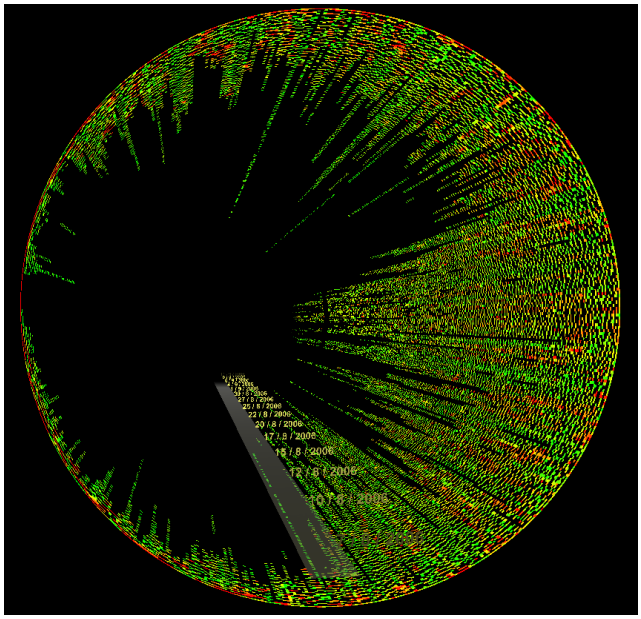


Fig. 7. Représentation de la glycémie pour 600 patients au cours de leur séjour dans un service de réanimatiom.

5. EVALUATION UTILISATEUR

5.1 Généralités sur le protocole utilisé

Nous avons réalisé une évaluation utilisateur afin de mieux quantifier l'intérêt de DataTube2. Dans notre cadre de la fouille visuelle de données, une telle évaluation consiste à définir des tâches à accomplir et les données correspondantes, à recruter des utilisateurs qui connaissent déjà les principes de la fouille de données, et à définir quelles versions de DataTube2 tester et quel logiciel "concurrent" utiliser.

Les tâches à accomplir ont été choisies parmi celles rencontrées couramment en fouille de données. Pour chacune d'elles, il a fallu également définir des données à utiliser ainsi qu'un critère permettant de quantifier la qualité de la réponse de l'utilisateur (en plus du temps de réponse). Nous avons choisi les trois tâches suivantes (voir les trois sections qui suivent) :

- la première tâche consiste à identifier le nombre de classes de variables présentes dans les données,
- la deuxième tâche consiste à détecter des variables dont le comportement temporel les isole des autres (des "outliers"),
- la troisième tâche consiste à trouver deux variables absolument identiques.

Nous avons recruté des utilisateurs ayant un minimum de formation en informatique, en Statistiques et en Fouille de données afin qu'aucune

incompréhension globale ne vienne perturber l'évaluation. Nous avons donc fait appel à 20 étudiants en Informatique de l'Université de Tours, âgés de 20 à 33 ans, et avec un niveau d'étude allant de bac+3 jusqu'au Doctorat.

Nous avons utilisé deux versions de DataTube2, avec ou sans BEA, afin de tester l'impact de la réorganisation sur la résolution des tâches précédentes. La stéréoscopie est utilisée dans les deux cas avec un écran 3D de 21 pouces. Afin d'éviter un apprentissage trop long des déplacements avec l'utilisation du SpacePilot (nous ne pouvons pas "bloquer" les utilisateurs pendant plus de 45 minutes), les déplacements sont restreints le long de l'axe du tube, ce qui doit être suffisant pour résoudre les tâches que nous avons définies.

Pour effectuer des comparaisons, nous avons cherché un logiciel qui soit déjà maîtrisé par les utilisateurs et qui utilise si possible une représentation matricielle 2D des données. Nous avons donc sélectionné un tableur, en utilisant les possibilités de coloration des cases. Ce tableur peut donc représenter la matrice des données en 2D avec des codes couleurs (voir figure 8(c)). L'utilisateur peut se déplacer en 2D et faire des zooms. La réorganisation n'étant pas une fonctionnalité du tableur celle-ci n'est pas utilisée.

Nous avons donc 3 méthodes à tester (DataTube2 avec ou sans BEA, tableur), et pour cela nous avons défini globalement le protocole suivant :

(1) L'utilisateur remplit un questionnaire préalable sur son niveau en Informatique, son expérience de la 3D (cinéma, jeux, écrans, matériel de navigation, programmation),

(2) On explique à l'utilisateur les principes de la fouille de données temporelles sur un exemple de données fournies par Google sur la grippe H1N1 (effectifs par région et en fonction du temps). Les trois tâches sont expliquées sur ces données ainsi que les trois représentations testées, DataTube2 avec ou sans BEA et le tableur. Nous nous assurons également que la personne perçoive correctement la stéréoscopie,

(3) Les trois tâches sont réalisées en mesurant le temps de réponse, le critère de qualité associé et le nombre de réponses exactes. Pour chacune d'elle, on choisit un seul logiciel à tester, selon une randomisation permettant, à la fin, d'avoir au moins 6 tests de chaque logiciel pour chaque tâche et de faire en sorte que chaque utilisateur ne teste jamais deux fois la même base ou deux fois le même logiciel,

(4) L'utilisateur répond à un questionnaire d'évaluation, jugeant de manière comparative le tableur et DataTube2, et plus généralement l'utilisation de DataTube2.

5.2 Détection du nombre de classes

Cette tâche consiste à faire trouver par l'utilisateur le nombre de classes de variables présentes dans les données. Pour obtenir le jeu de données correspondant, nous générons 3 variables temporelles différentes, et nous les dupliquons jusqu'à obtenir une visualisation avec 20 variables. Trois classes sont donc définies dans ces données avec respectivement 4, 6 et 10 variables (voir figure 8). Tous les membres d'une classe sont identiques afin de ne pas créer trop d'ambigüité ou de variabilité dans les réponses des utilisateurs, car la

classification est un processus subjectif. Le critère utilisé pour quantifier la réponse de l'utilisateur vaut 1 si la réponse donnée est "3" (réponse juste), puis 0.5 pour les réponses "2" ou "4", et 0 pour toutes les autres réponses.

Les résultats sont présentés dans la table III(a). Du point de vue de la qualité des résultats, on peut dire que les utilisateurs arrivent à bien résoudre cette tâche avec les trois méthodes, mais on note cependant un avantage pour DataTube2 où les utilisateurs commettent moins d'erreur qu'avec le tableur. La réorganisation avec BEA n'apporte sur cette tâche pas de bénéfice significatif en terme de qualité des résultats, par contre, le temps de réponse est clairement divisé par deux avec BEA. Si l'on tient compte des deux critères (qualité, temps), DataTube2 se comporte mieux que le tableur.

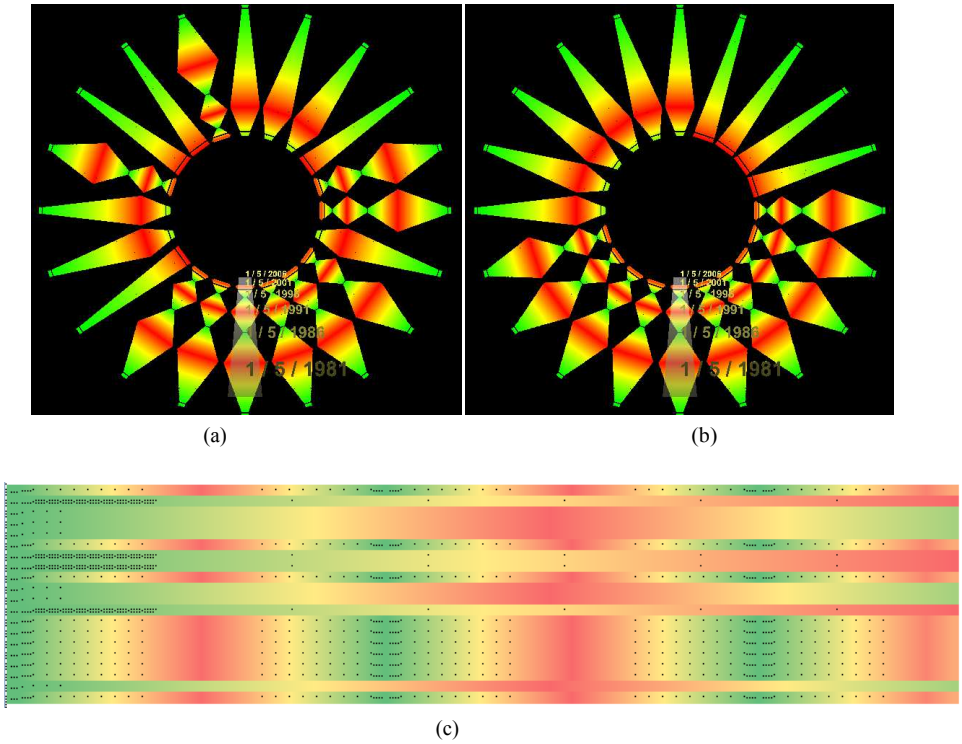


Fig. 8. Base de données utilisée pour le test sur la détection des classes de variables, avec en (a) la visualisation DataTube2 sans BEA, avec BEA en (b), et dans le tableur en (c).

5.3 Détection des variables aberrantes

Pour cette tâche, nous avons généré 17 variables au comportement "régulier" réparties en trois classes de 4, 5 et 8 variables. Au sein d'une même classe, toutes les variables sont identiques. Puis nous avons ajouté 3 variables au comportement "aberrant" par rapport aux trois classes précédemment définies.

Méthode	Temps	Qualité	Nb réponses exactes
DataTube2 sans BEA	44.0 (47.5)	1.00 (0.00)	7/7
DataTube2 avec BEA	19.8 (12.2)	0.83 (0.40)	5/6
Tableur	43.3 (35.6)	0.78 (0.39)	5/7

(a) identification du nombre de classes

Méthode	Temps	Qualité	Nb réponses exactes
DataTube2 sans BEA	33.6 (15,0)	0.66 (0.21)	1/6
DataTube2 avec BEA	37.7 (21.9)	0.80 (0.26)	4/7
Tableur	43.7 (16.4)	0.52 (0.28)	1/7

(b) détection des variables aberrantes

Méthode	Temps	Qualité	Nb réponses exactes
DataTube2 sans BEA	50.1 (27.8)	0.75 (0.43)	5/7
DataTube2 avec BEA	38.8 (41.5)	1.00 (0.00)	7/7
Tableur	105.3 (56.2)	0.25 (0.41)	1/6

(c) détection de deux variables identiques

Table III. Résultats obtenus par DataTube2 et le tableur pour les trois tâches (a) nombre de classes, (b) variables aberrantes, (c) deux variables identiques. Les valeurs sont en moyenne sur 6 ou 7 utilisateurs, les valeurs entre parenthèses sont les écarts-types. Les temps sont indiqués en seconde.

L'objectif de l'utilisateur est de détecter ces trois variables. Pour quantifier la qualité de la réponse, on définit un score qui vaut 1 si les trois variables sont trouvées par l'utilisateur, et sinon on enlève de cette valeur 1 des pénalités de 0.33 pour chaque variable aberrante non trouvée et de même pour chaque variable mentionnée mais qui n'était pas aberrante.

Dans la table III(b), on constate que cette tâche est plus difficile que la précédente, car il y a moins de réponses exactes. Le meilleur score de qualité est obtenu par DataTube2 avec BEA. En effet, la réorganisation a tendance à regrouper les variables des classes, puis à placer ensemble les variables aberrantes, ce qui permet de les identifier très nettement. Le tableur obtient le plus mauvais score de qualité. En observant plus finement les résultats, on constate que le tableur est la seule méthode pour laquelle une variable régulière a été mentionnée comme aberrante. En termes de temps de réponse, DataTube2 se comporte mieux que le tableur, mais pas de manière aussi significative que dans l'expérience précédente.

5.4 Détection des variables identiques

Nous générons cette fois une base de données avec 20 variables toutes différentes entre elles sauf un couple de variables totalement identiques. L'objectif de l'utilisateur est de trouver ces deux variables. Les résultats sont présentés dans la table III(c). En ce qui concerne la qualité, on constate qu'avec le tableur la tâche est très mal résolue par les utilisateurs. DataTube2 est donc nettement meilleur

que le tableur, et on constate un avantage avec l'utilisation de BEA car les deux variables identiques sont placées côte à côte. Les temps de réponse observés permettent d'aboutir à la même conclusion.

Question	Réponse pour tableur	Réponse pour DataTube2
Lecture des caractères	2.55 (1.14)	2.40 (0.80)
Se repérer les données	2.45 (0.88)	2.40 (1.09)
Facile à utiliser	4.75 (0.55)	4.05 (1.09)
Le plus intéressant	2.80 (1.23)	4.35 (0.87)
Le mieux adapté pour présentation	2.35 (0.98)	4.55 (0.68)

Table IV. Résumé des réponses données par les utilisateurs en ce qui concerne la comparaison entre le tableur et DataTube2. Les valeurs sont en moyenne sur 20 utilisateurs, les valeurs entre parenthèses sont les écarts-types.

Question	Réponse pour DataTube2
DataTube2 est frustrant ou satisfaisant	4.30 (0.92)
DataTube2 est ennuyeux ou stimulant	3.95 (0.60)
DataTube2 est rapide	3.95 (0.75)
DataTube2 est fiable	3.90 (0.78)

Table V. Résumé des réponses données par les utilisateurs en ce qui concerne DataTube2.

5.5 Questionnaire final et conclusion sur ces tests

Comme mentionné précédemment, le protocole se termine par un questionnaire d'évaluation. Pour toutes les questions, les réponses possibles vont de 1 à 5. Une première série de questions concerne la comparaison entre le tableur et DataTube2. Les résultats (voir table IV) montrent que les deux méthodes sont jugées de manière comparable en termes de lecture des informations, de positionnement dans les données et de facilité d'utilisation, mais ces résultats montrent aussi que DataTube2 est plus intéressant que le tableur notamment pour la présentation des données à d'autres personnes. On retrouve ici l'attrait des méthodes fondées sur la 3D quand il s'agit de présenter des informations, un point qui est crucial en fouille de données. En effet, dans les applications réelles, les résultats d'une analyse doivent généralement être présentés à des décideurs, et la forme de cette présentation a une grande importance.

Une deuxième série de questions a concerné DataTube2 seul (voir table V). Les réponses données par les utilisateurs montrent qu'ils jugent positivement DataTube2.

Globalement, l'évaluation utilisateur présentée ici confirme ce que nous avons pu relever dans les résultats de la section 4. DataTube2 est une visualisation facilement compréhensible qui améliore la résolution de tâche de fouille de données temporelles, à la fois en terme de temps et de qualité. Le coté "attrayant" de DataTube2 est également important à noter.

6. CONCLUSION

Dans cet article, nous avons tenté d'apporter une contribution à la visualisation interactive de grands volumes de données temporelles. Pour cela, nous avons étendu une méthode de visualisation existante mais n'ayant pas été testée dans des conditions réelles et avec des données en volume conséquent. Les extensions réalisées ont porté notamment sur l'amélioration de la visualisation et des interactions, sur l'utilisation d'un affichage stéréoscopique en réalité virtuelle, ainsi que sur la réorganisation des attributs par similarité. Nous avons essayé de rendre cet outil le plus complet et le plus opérationnel possible, ce qui nous a permis de l'appliquer à plusieurs bases de données réelles. En ce qui concerne les résultats, nous traitons des ensembles de données plus conséquents que les approches visuelles concurrentes, en proposant un regroupement des variables similaires ainsi qu'un passage intuitif d'une vue globale vers des vues détaillées. L'évaluation utilisateur a confirmé les bonnes propriétés de cette représentation.

Parmi les perspectives, nous souhaitons repousser les limites du nombre de données visualisées par amélioration des méthodes d'affichage. Egalement, l'affichage des couleurs et plus généralement le codage attributs/facettes peut être amélioré en proposant à l'utilisant des échelles non linéaires (pour la coloration par exemple). Ainsi, des méthodes de distorsion pourraient être appliquées sur les couleurs afin de faire apparaître des phénomènes qui sont invisibles à l'échelle globale. Nous testons aussi d'autres algorithmes de réorganisation pour améliorer les résultats de BEA et apporter différentes solutions à l'utilisateur. Cela concerne à la fois la définition de nouveaux algorithmes, d'autres mesures de similarité, par exemple pour des données symboliques avec des distances entre séquences, mais aussi la prise en compte d'une structure hiérarchique dans les variables temporelles. Pour ce dernier point, nous avons constaté plusieurs fois cette demande de la part de nos partenaires : pour les Log, il peut s'agir de la structure arborescente du site, pour les EEG, le positionnement de l'électrode sur le crâne, etc. DataTube2 pourrait donc être complétée par un affichage radial d'un arbre en son centre, à la demande de l'utilisateur, afin de matérialiser des groupes imbriqués de variables. De nouveaux algorithmes de réorganisation hiérarchiques devront être définis, car BEA ne traite que des données tabulaires.

Ces premiers résultats, soumis aux experts des différents domaines (logs Web et EEG), ont montré que la visualisation permettait une analyse rapide des données pour une personne non initiée. De plus, l'aspect 3D ajoute un aspect attractif pour la présentation des résultats à des clients ou demandeurs qui ne sont pas les analystes premièrement concernés. Notre but est maintenant de la développer pour la rendre encore plus intuitive et de la soumettre à un plus grand nombre d'utilisateurs pour traiter des données à plus grande échelle (logs web sur une ou plusieurs années et EEG concernant des études psychologiques en cours de réalisation). Nous sommes également en train de tester DataTube2 dans un cadre industriel concernant l'étude du comportement temporel de systèmes électroniques complexes.

REMERCIEMENTS

Nous tenons à remercier Lucie Angel, Badiaa Bouazzaoui et Michel Isingrini

(équipe Vieillissement et Mémoire du laboratoire Langage, Mémoire et Développement Cognitif, UMR 6215, Université de Poitiers et Université de Tours) pour nous avoir fourni les données d'EEG, ainsi que Pierre Kalfon et Gilles Rafalli (Service de Réanimation de l'Hôpital de Chartres, et société LK2) pour les données sur la glycémie. Nous remercions Simon Assani, Romain Lucas et Florian Sureau pour leur aide dans l'implémentation de DataTube2.

RÉFÉRENCES

Ankerst, M. 2000. Visual data mining. Ph.D. thesis, Faculty of Mathematics and Computer Science, University of Munich. ISBN 3-89825-201-9.

Ankerst, M. 2001. Visual data mining with pixel-oriented visualization techniques. *Proceedings of the ACM SIGKDD Workshop on Visual Data Mining.*

Ankerst, M., Jones, D., Kao, A., and Wang, C. 1996. Datajewel: Tightly integrating visualization with temporal data mining. *ICDM Workshop on Visual Data Mining*.

Ankerst, M., Keim, D., and Kriegel, H. 1996. Circle segments: A technique for visually exploring large multidimensional data sets. *Proc. IEEE Visualization'96, Hot Topics 96.*

Antunes, C. and Oliveira, A. 2001. Temporal data mining: An overview. *KDD Workshop on Temporal Data Mining.*

Azzag, H., Picarougne, F., Guinot, C., and Venturini, G. 2005. Vrminer: A tool for multimedia database mining with virtual reality. *Processing and Managing Complex Data for Decision Support.,* 318–339.

Benabdeslem, K., Bennani, Y., and Janvier, E. 2002. Visualization and analysis of web navigation data. *In Springer ICANN: International Conference on Artificial Neural Networks*. 486–491.

Bender-deMoll, S. and McFarland, D. 2006. The art and science of dynamic network visualization.
Journal of Social Structure, 7(2).

Bertin, J. 1977. La graphique et le traitement graphique de l'information. *Nouvelle Bibliothèque Scientifique..*

Carlis, J. and Konstan, J. 1998. Interactive visualization of serial periodic data. *Proceedings of the 11th annual ACM symposium on User interface software and technology,* 29–38.

Chi, E., Pitkom, J., Ma&inlay, J., Pirolli, P., Gossweiler, R., and Card, S. 1998. Visualizing the evolution of web ecologies. *Proceedings of the Human Factors in Computing Systems,* 400–407.

Chi, E., Pitkow, J., Mackinlay, J., Pirolli, P., Gossweiler, R., and Card, S. 1998. Visualizing the evolution of web ecologies. *Proceedings of the Human Factors in Computing Systems,* 400–407.

Climer, S. and Zhang, W. 2006. Rearrangement Clustering: Pitfalls, Remedies, and Applications. *The*
Journal of Machine Learning Research 7, 919–943.

Cugini, J. and Scholtz, J. 1999. VISVIP: 3D visualization of paths through web sites. *Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on,* 259–263.

Daassi, C., Dumas, M., Fauvet, M., Nigay, L., and Scholl, P. 2000. Visual exploration of temporal object databases. *proc. of BDA00 Conference,* 24–27.

Fay, S., Isingrini, M., Ragot, R., and Pouthas, V. 2005. The effect of encoding manipulation on word-stem cued recall: an event-related potential study. *Cognitive brain research 24*, 3, 615–626.

Francis, B. and Pritchard, J. 2003. Visualisation of historical events using lexis pencils. *Case Studies of Visualization in the Social Sciences 30.*

Hackstadt, S. T. and Malony, A. D. 1994. Visualizing parallel program and performance data with ibm visualisation data explorer. M.S. thesis.

Hébrail, G. and Debregeas, A. 1998. Interactive interpretation of kohonen maps applied to curves.

Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining. AAAI press, Menlo Park, 179–183.

Jain, A., Murty, M., and Flynn, P. 1999. Data clustering: a review. *ACM Computing Surveys (CSUR) 31*, 3, 264–323.

Kandogan, E. 2000. Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. *IEEE Symposium on Information Visualization 2000*, 4–8.

Keim, D., Ankerst, M., and Kriegel, H. 1995. Recursive pattern: A technique for visualizing very large amounts of data. *Proceedings of the 6th conference on Visualization'95*, 279–286.

Kizhakke, V. 2000. MIR: A tool for visual presentation of web access behavior. Ph.D. thesis, Uiversity of Florida.

McCormick, W., Schweitzer, P., and White, T. 1972. Problem decomposition and data reorganization by a clustering technique. *Operations Research 20*, 5, 993–1009.

Minar, N. and Donath, J. 1999. Visualizing the crowds at a web site. *CHI'99 Late Breaking Papers*.

Minard, C. 1861. Carte figurative des pertes successives en hommes de l'armée française dans la campagne de russie 1812-1813.

Muller, W. and Schumann, H. 2003. Visualization methods for time-dependent data-an overview.
Simulation Conference, 2003. Proceedings of the 2003 Winter 1, 737–745.

Otjacques, B. 2008. Techniques de visualisation des informations associées à une plate-forme de coopération. Ph.D. thesis, Facultés Universitaires Notre-Dame de la Paix, Institut d'Informatique, Namur, Belgique.

Pitkow, J. and Bharat, K. 1994. WEBVIZ: A Tool for World-Wide Web Access Log Visualization. *Proceedings of the First International World Wide Web Conference*, 271–277.

Scullin, W. H., Kwan, T. T., and Reed, D. A. 1995. Real-time visualization of world wide web traffic, Symposium on visualizing time-varying data.

Skog, T. and Holmquist, L. 2000. Webaware: Continuous visualization of web site activity in a public space. *Poster at CHI'2000*.

Sureau, F., Bouali, F., and Venturini, G. 2009. Optimisation heuristique et génétique de visualisations 2d et 3d dans olap : premiers résultats. *RNTI, 5ème journées francophones sur les entrepôts de données et l'analyse en ligne (EDA'09)*, 62–75.

Symanzik, J., Cook, D., Kohlmeyer, B. D., and Cruz-Neira, C. 1996. Dynamic statistical graphics in the cave virtual reality environment. In *Proc. Dynamic Statistical Graphics Workshop*. 41–47.

Theron, R. 2006. Hierarchical-Temporal Data Visualization Using a Tree-Ring Metaphor. *Lecture Notes in Computer Science 4073/2006,* 70–81.

van Wijk, J. and van Selow, E. 1999. Cluster and calendar based visualization of time series data. *Proceedings of IEEE Symposium on Information Visualization*, 4–9.

Wattenberg, M. 2002. Arc diagrams: visualizing structure in strings. Info*rmation Visualization, 2002.*
INFOVIS 2002. IEEE Symposium on, 110–116.

Weber, M., Alexa, M., and Muller, W. 2001. Visualizing time-series on spirals. *Information Visualization, 2001. INFOVIS 2001. IEEE Symposium on,* 7–13.

Fatma Bouali est Maître de Conférences en Informatique à l'Université de Lille 2 et membre du Laboratoire d'Informatique de l'Université François Rabelais de Tours. Elle s'intéresse à la fouille visuelle de données et plus particulièrement dans le cadre des données multidimensionnelles (OLAP) et des données complexes.

Fréderic Plantard est Ingénieur en Informatique et a obtenu son Master de Recherche en 2009 au Laboratoire d'Informatique de l'Université François Rabelais de Tours. Il travaille actuellement dans le secteur privé comme Ingénieur en Réalité Augmentée à Paris. Ses centres d'intérêt scientifiques sont la Réalité Virtuelle, les algorithmes biomimétiques et les mathématiques.

Amina Bouseba est ingénieur en Informatique et a obtenu son Master Recherche en 2010 au Laboratoire d'Informatique de l'Université François Rabelais de Tours. Elle est actuellement en Master 2 SIAD (Systèmes d'information et analyse décisionnelle) à l'Université de Tours.

Gilles Venturini est Professeur en Informatique et membre du Laboratoire d'Informatique de l'Université François Rabelais de Tours. Il dirige l'équipe Fouille visuelle de données et algorithmes biomimétiques. Ses centres d'intérêt portent de manière générale sur l'interface entre l'expert du domaine et les outils de fouille de données, et plus précisément sur les visualisations interactives et en réalité virtuelle de données complexes dans le domaine biomédical, et sur l'acquisition de données 3D. Il est membre du comité de pilotage de l'association Extraction et Gestion des Connaissances. Il est président de la Société Francophone de Classification.