



**HAL**  
open science

## Visualisation dynamique de classification d'images semi-supervisée incrémentale

Pierrick Bruneau, Fabien Picarougne, Marc Gelgon

► **To cite this version:**

Pierrick Bruneau, Fabien Picarougne, Marc Gelgon. Visualisation dynamique de classification d'images semi-supervisée incrémentale. joint meeting of the Société Francophone de Classification and the Classification and Data Analysis Group of the Italian Society of Statistics (SFC-CLADAG'10), Jun 2010, St Denis de la Réunion, France. pp.13-16. hal-00581262

**HAL Id: hal-00581262**

**<https://hal.science/hal-00581262>**

Submitted on 30 Mar 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Visualisation dynamique de classification d'images semi-supervisée incrémentale

Pierrick Bruneau\*, Fabien Picarougne\*  
Marc Gelgon\*

\*LINA, Polytech'Nantes, rue Christian Pauc, BP 50609, 44306 Nantes Cedex3  
{prenom.nom}@univ-nantes.fr

**Résumé.** Dans le contexte de l'analyse de flux réseaux en temps réel, il peut être difficile pour un utilisateur d'interpréter le flux de résultats provenant d'un algorithme de classification incrémentale. Nous proposons dans ce papier une nouvelle méthode de visualisation dynamique de classification permettant de suivre un résumé de l'évolution temporelle des clusters produits par l'algorithme. À chaque instant, l'état de la classification est représenté par un graphe complet dont les nœuds sont modélisés par les clusters calculés par l'algorithme de classification et les arcs valués par la distance inter-clusters. La position de chaque nœud est calculée en utilisant un algorithme basé sur un modèle de force et ressorts prenant en compte l'évolution des paramètres du graphe.

## 1 Introduction

La recherche d'images en fonction du contenu est un problème abondamment traité dans la littérature (voir Datta et al. (2005)). Une des pistes explorées dans ce domaine concerne les aspects de propriété intellectuelle et la recherche de contenu possédant des attributs graphiques communs. Ce problème est communément traité par l'utilisation d'algorithmes de classification. Mais lorsque les critères de recherche ne peuvent pas être bien définis dès le départ, comme par exemple dans le cas de problématiques d'interception légale, il peut être judicieux de faire intervenir l'humain dans le processus de classification (voir Carbonetto et al. (2008)).

Dans ce papier, nous nous plaçons dans le contexte d'analyse de grands flux d'images circulant sur un réseau ; les données provenant du projet ANR SAFIMAGE. La volumétrie des données échangées dans cette situation impose l'utilisation d'algorithmes de classification incrémentaux et de prendre en compte une intervention humaine dans le processus de décision.

Cependant, dans ce contexte applicatif d'analyse du contenu de flux réseaux en temps réel, il peut être difficile pour un utilisateur d'interpréter le flux de résultats provenant d'un algorithme de classification incrémental. Afin d'aider l'utilisateur dans sa tâche, nous proposons dans ce papier une nouvelle méthode de visualisation dynamique de classification permettant de suivre un résumé de l'évolution temporelle des clusters produits par l'algorithme et d'interagir avec ce dernier à partir de la représentation visuelle des classes. Nous ne traiterons pas

ici des détails de l'algorithme de classification utilisé mais considèrerons qu'il est possible de récupérer des informations sur le contenu des groupes formés à chaque étape de la classification et nous ferons l'hypothèse qu'il existe une formulation de distance entre chaque groupe d'images identifiés par l'algorithme de classification.

## 2 Modélisation et principe de visualisation

Dans le contexte d'application envisagé, le processus de classification incrémental produit des groupes qui évoluent dans le temps. Nous devons donc construire une représentation qui puisse montrer à l'utilisateur ce qui a changé à chaque itération de l'algorithme.

Plusieurs méthodes ont été proposées dans la littérature afin de visualiser des séries temporelles. La plupart d'entre elles (Bender-deMoll et McFarland (2006); Kang et al. (2007)) produisent une nouvelle visualisation à chaque changement d'état du modèle sous-jacent et montrent la dynamique d'évolution en visualisant un morphing entre les représentations au temps  $t$  et au temps  $t + 1$ . Mais dans ce cas, le succession des étapes du processus de visualisation mènent à une discontinuité visuelle dans la représentation des classes.

La méthode de visualisation que nous proposons se base sur une représentation par un graphe dynamique qui évolue dans le temps. Nous avons choisis dans un premier temps de ne représenter visuellement que les clusters formés par l'algorithme de classification afin de ne pas surcharger la quantité d'information transmise à l'utilisateur. Chaque cluster est symbolisé par un résumé des informations qui le compose et nous cherchons principalement à visualiser les proximités entre les différents groupes.

À chaque instant, l'état de la classification est représenté par un graphe complet dont les nœuds sont modélisés par les clusters calculés par l'algorithme de classification et les arcs valués par la distance inter-clusters. Le nombre de nœuds du graphe et la valeur associé à chaque arc évolue en permanence au cours de la classification. L'ensemble est projeté dans un espace de visualisation à trois dimensions dans laquelle la distance entre les nœuds est fonction de la valeur des arcs les reliant.

Le plongement en trois dimensions est réalisé ici par un algorithme de représentation de graphe basé sur un système de force et ressort (voir Fruchterman et Reingold (1991)). Les algorithmes de classification que nous proposons produisent un nombre limité de groupes à visualiser (moins de 30) et ce type d'algorithme est connu pour produire efficacement de très bons résultats pour des graphes non orientés de petites à moyenne taille (jusqu'à 50 ou 100 nœuds). Ils produisent une approximation physique qui peut être facilement interprétée par un humain et dont les nœuds sont généralement bien distribués dans l'espace tout en préservant les symétries du graphe. Nous utilisons un modèle qui considère chaque arc du graphe à visualiser comme un ressort et chaque nœud comme une particule électriquement chargée. Les forces s'exerçant sur le graphe sont alors simulées comme dans un système physique. À chaque itération, nous calculons la somme de chaque force du système en utilisant la force de Hooke pour les arêtes et la force de Coulomb pour les nœuds. La première force tend à maintenir les distances requises entre chaque nœud tandis que la force de Coulomb écarte le plus possible les nœuds entre eux.

Cet algorithme permet de visualiser l'évolution de la population de nœuds du graphe de manière continue en se basant sur des principes de déplacement physiques. Ces principes sont

habituellement rencontrés dans la nature et sont par conséquent facile à comprendre et à prédire pour un humain.

### 3 Algorithme de placement

Afin de permettre à l'utilisateur d'interpréter et de comprendre le principe de formation des clusters, nous avons choisis d'utiliser une version incrémentale en trois dimensions de l'algorithme initial de force et ressorts. Ce type d'algorithme a déjà été utilisé afin de visualiser interactivement des graphes (voir McCrickard et Kehoe (1997)).

À chaque itération de l'algorithme de visualisation, la force de Coulomb est calculée pour chaque nœud afin de les éloigner les uns des autres. Cette force est d'autant plus forte que la distance séparant les nœuds est faible. La force de Hooke est également évaluée pour chaque arc, tendant à faire respecter la longueur à vide de chaque ressort : attraction des nœuds reliés à l'arc si celui-ci est trop grand et répulsion dans le cas contraire. La somme des force engendre ensuite un déplacement des nœuds dans l'espace de représentation tri-dimensionnelle, tendant à converger vers un équilibre global des forces du système.

La convergence du système permet à l'utilisateur de voir naturellement l'évolution du dessin du graphe et par conséquent d'interpréter la construction des clusters. Cette représentation permet également de proposer des mécanismes d'interaction qui permettent à l'utilisateur d'interagir avec elle.

Nous avons choisis de représenter les nœuds du graphe par les images les plus représentatives de chaque cluster. Cela permet de donner à l'utilisateur une idée des caractéristiques visuelles principales des données de chaque groupe sans surcharger la visualisation globale.

La figure 1 montre une représentation statique de la visualisation produite par notre application. On peut noter que les images représentant les groupes produits par l'algorithme de classification sont toujours affichés de manière à faire face à l'utilisateur (billboard 3D) afin de rendre leur lecture plus facile.

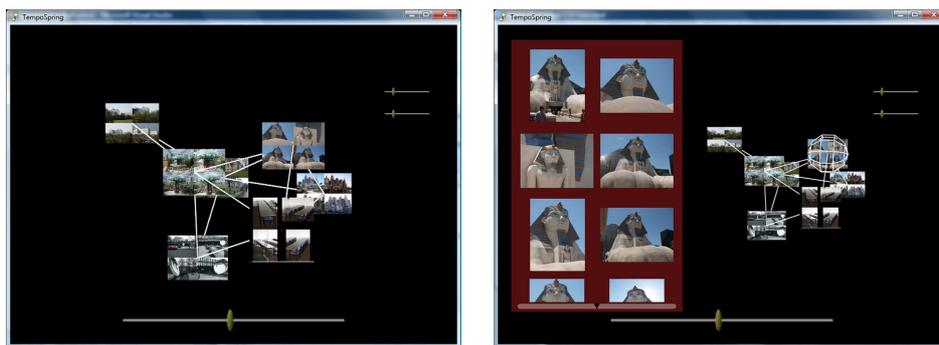


FIG. 1 – Capture d'écran de la classification montrant l'interface de manipulation du graphe à gauche et visualisant interactivement le contenu des clusters à droite.

## 4 Amélioration de la visualisation et conclusion

Plusieurs améliorations ont été introduites afin d'enrichir la visualisation produite. 1) Le centre de gravité du graphe est centré à chaque itération dans la fenêtre de visualisation. Ce changement de position se fait en contrôlant l'accélération subie par chaque nœud afin de ne pas engendrer un « saut » dans la visualisation. 2) Afficher un graphe complet peut également diminuer la lisibilité du graphe. Nous avons donc choisi de ne pas prendre en compte dans la modélisation du graphe les arcs dont la distance inter-cluster était trop grande. Ce seuil d'affichage est déterminé par l'utilisateur à l'aide d'un slider situé dans la partie inférieure de la fenêtre d'affichage ; le système restreignant ce seuil afin de préserver la connexité du graphe. 3) La perception de la profondeur en 3D est facilitée par l'utilisation de systèmes de visualisation stéréoscopique, autorisant la manipulation aisée et rapide des structures tri-dimensionnelles. 4) Enfin, nous permettons à l'utilisateur d'interagir avec la visualisation afin d'obtenir des détails sur les clusters formés et d'apporter des modifications à ces clusters dans le cadre d'un contexte semi-supervisé (fusion de clusters, étiquetage d'individus, ...).

## Références

- Bender-deMoll, S. et D. A. McFarland (2006). The art and science of dynamic network visualization. *Journal of Social Structure* 7(2).
- Carbonetto, P., G. Dorko, C. Schmid, H. Kück, et N. de Freitas (2008). Learning to recognize objects with little supervision. *International Journal of Computer Vision* 77(1-3), 219–237.
- Datta, R., J. Li, et J. Z. Wang (2005). Content-based image retrieval : approaches and trends of the new age. *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, 253–262.
- Fruchterman, T. M. J. et E. M. Reingold (1991). Graph drawing by force-directed placement. *Softw. Pract. Exper.* 21(11), 1129–1164.
- Kang, H., L. Getoor, et L. Singh (2007). Visual analysis of dynamic group membership in temporal social networks. *SIGKDD Explor. Newsl.* 9(2), 13–21.
- McCrickard, D. S. et C. M. Kehoe (1997). Visualizing search results using sqwid. pp. 51–60. ACM Press.

## Summary

In the context of the analysis of real time network stream, it may be difficult for a user to interpret the stream of results from an incremental clustering algorithm. We propose in this paper a new dynamic visualization method of clustering letting the user to follow a summary of the temporal evolution of clusters produced by the clustering algorithm. At each time, the clustering state is represented by a complete graph where the nodes are modeled by clusters calculated by the clustering algorithm and the arcs weighted by the distance between clusters. The position of each node is computed using a spring based algorithm that takes into account the evolution of the parameters of the graph and the position of the nodes at the previous iteration.