

Applications des graphes en traitement d'images

Salim Jouili, Salvatore Tabbone

► **To cite this version:**

Salim Jouili, Salvatore Tabbone. Applications des graphes en traitement d'images. Y. Boudabbous and N. Zaguia. International Conference on Relations, Orders and Graphs: Interaction with Computer Science - ROGICS'08, May 2008, Mahdia, Tunisie. pp.434-442, 2008. <inria-00334793>

HAL Id: inria-00334793

<https://hal.inria.fr/inria-00334793>

Submitted on 28 Oct 2008

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.

Applications des graphes en traitement d'images*

Salim Jouili, Salvatore Tabbone
LORIA - UMR 7503
Université de Nancy 2
Campus scientifique - BP 239
54 506, Vandœuvre -les-Nancy Cedex, France
{salim.jouili,tabbone}@loria.fr

Résumé

Les graphes sont des outils de représentation des données très puissants et universels, utiles dans divers domaines des sciences notamment le traitement d'images, la reconnaissance des formes ou la vision par ordinateur. Dans ces domaines, la mesure de similarité entre les objets est souvent une phase importante. Dans le cadre d'une représentation sous forme de graphe cette phase se traduit par un appariement des graphes. Dans cet article nous passons en revue un ensemble de travaux basé sur la théorie des graphes appliquée aux traitements d'images. Notamment les approches de représentation en graphe et les différentes mesures de similarité de graphes. Également, nous présentons quelques applications des graphes dans l'analyse d'images et la recherche par le contenu.

Mots-clefs : Graphe, Traitement d'images, Mesure de similarité, Appariement de graphes.

1 Introduction

La représentation sous forme de graphe est utilisée dans différentes applications d'analyse d'images. Dans ces applications, l'image est traitée pour produire un graphe représentant les composants et les relations entre eux. Les nœuds et les arcs peuvent avoir des étiquettes (labels). Des informations auxiliaires sont enregistrées comme attributs associés aux nœuds et/ou aux arcs. La distinction entre les étiquettes et les attributs se fait comme suit : les étiquettes font parties de la structure du graphe et elles doivent s'apparier dans l'ordre pour que les graphes soient considérés isomorphes. En contre partie, les attributs contiennent des informations qui ne font pas parties de la structure du graphe. Les valeurs des attributs sont ignorées lors du test de similarité entre des graphes [7].

Cette représentation structurelle puissante a prouvé sa flexibilité en traitement d'une grande variété de types d'images (les anciens documents, les plans électriques et architecturaux, les images naturelles, les images médicaux ...). Les travaux, qui ont fait recours aux graphes, ont visé une représentation qui préserve l'information topographique de l'image ainsi que les relations entre les composantes. Actuellement, plusieurs approches dérivées des graphes pour l'analyse d'images ont été proposées notamment pour les systèmes de recherche par contenu, la segmentation et l'indexation.

Le fil conducteur de ce travail, est suggéré par la question suivante : *Où en est-on avec l'application des graphes dans l'analyse d'images ?* Dans ce contexte, nous explorons dans cet article des travaux sur la représentation sous forme de graphe, ensuite nous détaillons quelques travaux sur les mesures de similarité entre les graphes développées par des traiteurs d'images. Enfin, nous passerons en revue des applications des graphes dans le domaine de l'analyse d'images.

*Ce travail est partiellement financé par le projet Navidomass de l'Agence Nationale de la Recherche référencé sous le numéro ANR-06-MCDA-012 et par la région Lorraine.

2 Notions des graphes

2.1 Définitions

Soit E un ensemble, on écrit $X \subseteq E$ si X est un sous-ensemble de E , on écrit $X \subset E$ si X est un sous-ensemble propre de E (X est un sous-ensemble de E et $X \neq E$). On appelle \bar{X} l'ensemble complémentaire de X dans E avec $\bar{X} = E \setminus X$. Soit E un ensemble fini, on note par $\|E\|$ le nombre des éléments de E . On note par 2^E l'ensemble composé par tous les sous-ensembles de E .

On définit un **graphe** par le couple (V, E) où V est un ensemble fini et E une relation binaire dans E ($E \subseteq V \times V$), chaque élément dans V est appelé nœud. On note aussi par E la fonction de $V \rightarrow 2^V$ tels que $\forall x \in V, E(x) = \{y \in V \mid (x, y) \in E\}$. Si $y = V(x)$, y est adjacent à x . Soit la fonction E^* tels que $x \in V, E^*(x) = V(x) \setminus x$. Soit $X \subseteq V$, on définit $E(X) = \bigcup_{x \in X} E(x)$, et $E^*(X) = E(X) \setminus X$. Si $y \in E(X)$ on dit que y est adjacent à X . Soient $X, Y \subseteq V$ et $E(X) \cap Y = \emptyset$, alors Y est adjacent à X .

Soit G un graphe avec $G = (V, E)$ et $X \subseteq V$, on définit un sous-graphe de G induit par X par le graphe $G_X = (X, E \cap [X \times X])$. Dans ce cas, G_X est un sous-graphe de G . Soient les deux graphes $G = (V, E)$ et $G' = (V', E')$, G et G' sont isomorphes s'il existe une bijection f de $E \rightarrow E'$ tels que, $\forall x, y \in V, y \in E(x)$ si et seulement si $f(y) \in E'(f(x))$.

Soit G un graphe avec $G = (E, V)$ G est :

réflexif : $\forall x \in V, (x, x) \in E$.

irréflexif : $\forall x \in V, (x, x) \notin E$.

symétrique : $\forall x, y \in V, (x, y) \in E \Rightarrow (y, x) \in E$.

asymétrique : $\forall x, y \in V, (x, y) \in E \Rightarrow (y, x) \notin E$ (si G est asymétrique, G est irréflexif).

antisymétrique : $\forall x, y \in V, (x, y) \in E$ et $(y, x) \in E \Rightarrow x = y$ (si G est asymétrique, G est aussi antisymétrique).

transitif : $\forall x, y \in V, (x, y) \in E, (y, z) \in E \Rightarrow (x, z) \in E$.

Un graphe relationnel attribué (Attributed Relational Graph) : est un graphe où chaque nœud ($\forall x \in V$) est associé à un vecteur d'attributs $A_v \subset \mathbb{R}_n$ et chaque arc ($\forall (x, y) \in E$) est associé à un vecteur d'attributs $A_e \subset \mathbb{R}_m$, avec n et $m \in \mathbb{N}$. Bien qu'on définisse les attributs comme de nombres réels, ce type de graphe n'est pas limité au nombre réel.

2.2 Appariements de graphes

Afin de comparer deux graphes, il est nécessaire de mettre en correspondance leurs sommets, c-à-d de définir un appariement (matching) entre ces graphes.

Étant donné deux graphes $G = (V, E)$ et $G' = (V', E')$, un appariement multivoque m entre G et G' est une relation entre V et V' , c-à-d $m \subseteq V \times V'$. Sans perte de généralité, il est supposé que les ensembles des nœuds de G et de G' sont disjoints ($V \cap V' = \emptyset$).

Nœuds appariés à un sommet : Étant donné un appariement m de deux graphes G et G' , $m(v)$ est l'ensemble des nœuds appariés au nœud v . Plus formellement :

$$\begin{aligned} \forall v \in V, m(v) &= \{v' \in V' \mid (v, v') \in m\} \\ \forall v' \in V', m(v') &= \{v \in V \mid (v, v') \in m\} \end{aligned}$$

Quand aucune contrainte n'est ajoutée à l'appariement, c-à-d quand chaque sommet peut être apparié à zéro, un, ou plusieurs nœuds (chaque nœud pouvant être apparié à un ensemble de nœud de l'autre graphe), l'appariement est dit multivoque. Cependant, il est possible d'ajouter des contraintes sur le nombre de sommets avec lesquels un nœud peut être apparié. Cela nous amène à définir les appariements fonctionnels, applicatifs, univoques, injectifs et bijectifs. Étant donnés deux graphes $G = (V, E)$ et $G' = (V', E')$, un appariement $m \subseteq V \times V'$ est :

- un appariement fonctionnel de G dans G' si m apparie chaque sommet du graphe G avec au plus un sommet du graphe G' : $\forall v \in V, \|m(v)\| \leq 1$
- un appariement applicatif de G dans G' si m apparie chaque sommet du graphe G avec exactement un sommet du graphe G' : $\forall v \in V, \|m(v)\| = 1$
- un appariement univoque entre G et G' si m apparie chaque sommet du graphe G et G' avec au plus un sommet : $\forall v \in V, \|m(v)\| \leq 1 \wedge \forall v' \in V', \|m(v')\| \leq 1$

- un appariement injectif de G dans G' si m apparie chaque sommet du graphe G avec un sommet différent de G' : $\forall v \in V, \|m(v)\|=1 \wedge \forall v' \in V', \|m(v')\| \leq 1$
- un appariement bijectif entre G et G' si m apparie chaque sommet du graphe G (respectivement de G') avec un sommet différent de G' (respectivement de G) : $\forall v \in V, \|m(v)\|=1 \wedge \forall v' \in V', \|m(v')\|=1$

Arcs appariés : Soit m un appariement des sommets de deux graphes $G=(V,E)$ et $G'=(V',E')$, un arc $(u,v) \in E$ est dit apparié à un autre arc $(u',v') \in E'$ si et seulement si $\{(u,u'), (v,v')\} \subseteq m$. Comme pour le nœuds l'ensemble des arcs appariés à l'arc (u,v) par l'appariement m est noté $m(u,v)$:

$$\forall (u,v) \in E, m(u,v) = \{(u',v') \in E' \mid u' \in m(u), v' \in m(v)\}$$

$$\forall (u',v') \in E', m(u',v') = \{(u,v) \in E \mid u \in m(u'), v \in m(v')\}$$

3 Représentation par des graphes

Les images sont généralement segmentées en régions. Chaque sommet du graphe représente alors une région de l'image et les arcs représentent les différentes relations possibles entre les régions (connectivité, distance...). L'intérêt principal de l'utilisation des graphes pour la représentation d'images est l'intégration d'informations spatiales dans la modélisation. En effet les représentations classiques telles que les histogrammes de couleurs, les descripteurs de textures, etc... ne donnent aucune information sur la façon dont les régions d'intérêt de l'image sont agencées. Au contraire, la représentation par les graphes permet de décrire la structure de l'image comme la façon dont les régions sont disposées les une par rapport aux autres. En outre, selon les types de relations choisies, cette représentation en graphe peut être invariante à certaines transformations telles que les rotations de l'image ou les translations de certaines parties de l'image.

Comme exemple de cette représentation sous forme de graphe Horst Bunke en [9] convertit un diagramme de circuit en un graphe avec la représentation des lignes du diagramme de circuit : chaque nœud du graphe représente un point final d'une ligne, angle, ou intersection, et les attributs du nœud enregistrent les coordonnées (x,y) de cette caractéristique. L'étiquette d'un nœud est le degré du nœud (1 pour un point final, 2 pour un angle, 3 pour une intersection en \top). En raison de cette définition des étiquettes des nœuds, l'isomorphisme de sous-graphes apparie toujours un nœud de point final avec un autre nœud de point final, jamais un avec un nœud angle ou intersection. Cependant, les coordonnées (x,y) sont ignorées durant l'isomorphisme de sous-graphe, parce que ces coordonnées sont enregistrées comme attributs.

D'autres travaux (détaillés dans [24, 1, 4, 25, 3, 30, 29, 15]) de représentation sous forme structurelle (i.e. graphes) des images ou des documents graphiques utilisent différentes méthodes (les méthodes varient selon les caractéristiques des données manipulées). Par exemple, dans [24] l'auteur manipule des lettrines issues de documents anciens, il procède par une segmentation en différentes couches d'informations de ces lettrines afin d'obtenir "*des couches d'informations de zones homogènes*". Chaque région homogène de la lettrine sera associée à un nœud du graphe avec deux attributs la taille et la forme, et chaque arc entre les nœuds à deux attributs l'angle et la distance. Dans [30], les graphes sont utilisés pour légender (étiqueter) les images par des mots-clés automatiquement. Un graphe principale est construit à partir d'un ensemble des images déjà étiquetées et segmentées. Ce graphe est un graphe de 3-niveaux (Fig.1), avec un niveau de nœuds images (les i dans la Fig.1), un niveau de nœuds mot-clés (les t dans la Fig.1) et un niveau de nœuds régions d'image (les r dans la Fig.1). Les connections entre les nœuds sont basées soit sur une relation de similarité soit sur une relation de co-occurrence. Pour capturer la corrélation d'un attribut croisé, pour chaque image, l'auteur soumet un arc entre le nœud image et les nœuds attributs valeurs associés à l'image. Cette liaison est appelée IAVlinks ("*image-attribute-value*" links).

La plupart des travaux qui utilisent les graphes comme modèle de représentation notamment pour les documents graphiques, ont pour objectif la définition de la mesure de similarité entre les documents pour les comparer. La section prochaine présente des mesures de similarités élaborées soit simplement pour comparer deux documents, soit pour être au cœur de systèmes d'indexation ou de classification basés sur les graphes.

4 Mesure de similarité des graphes

La structure des graphes est caractérisée principalement par la flexibilité, l'universalité et l'utilisation dans des domaines d'applications variés. Ce fait a conduit au développement de plusieurs mesures de similarité

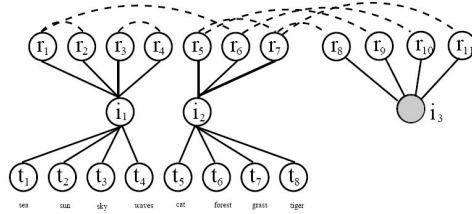


FIG. 1 – Un graphe à 3-niveaux [30]

pour les graphes qui sont optimisées pour différents types de graphes et différentes applications. Ces mesures de similarités de graphe diffèrent selon le type des graphes qu’elles traitent et qu’elles prennent ou non en compte les informations des attributs. Mais la plupart de ces mesures ont une caractéristique commune qui consiste en quelque sorte à utiliser des opérations d’édition (*edit operations*). Le principe est de définir l’effort nécessaire pour rendre les graphes identiques. Cet effort est calculé par le nombre d’opérations primitives qui sont nécessaires pour rendre les graphes identiques. Dans la suite nous détaillerons les mesures les plus connues dans la littérature.

4.1 La distance d’édition

La distance d’édition pour les graphes est l’extension de la distance d’édition pour les chaînes des caractères (ou distance de Levenshtein) [35] aux graphes. La distance d’édition est une mesure très commune de similitude pour les graphes et des variantes ont été proposés avec succès dans beaucoup de domaines d’application telle que par exemple la reconnaissance des visages [36]. La distance d’édition entre deux graphes est le nombre minimum des opérations d’édition qui sont nécessaires pour transformer un graphe en un autre. Les principales opérations d’édition sont, la suppression ou l’insertion des nœuds ou arcs, et le changement des attributs des nœuds ou des arcs.

Pour calculer la distance entre deux graphes Robles-Kelly [34] propose une méthode de conversion d’un graphe à une séquence de chaînes de caractères afin d’appliquer des techniques de mesure de la distance d’édition de chaîne de caractères. La problématique principale de cette mesure est la détermination du coût minimal des opérations d’édérations effectuées. Les solutions proposées face à cette problématique sont dans la majorité probabilistes [28] [34]. Papadopoulos et Manolopoulos [31] propose une autre solution en réduisant les opérations possibles (cf section suivante).

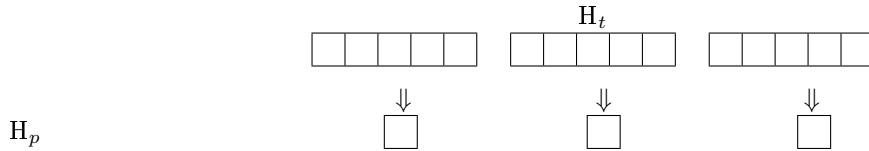
4.2 La mesure de Papadopoulos et Manolopoulos

Papadopoulos et Manolopoulos [31] présentent une mesure de similarité pour les graphes qui est aussi basée sur le concept des opérations d’édition. Ils proposent trois opérations primitives différentes, qui sont l’insertion de nœuds, la suppression de nœuds et la mise à jour de nœuds. Pendant que les opérations de suppression et d’insertion ont des significations évidentes, l’opération de mise à jour est nécessaire pour insérer ou supprimer les arcs incidents pour un nœud. En plus, Papadopoulos et Manolopoulos introduisent la séquence de degrés d’un graphe qui est la donnée en ordre décroissante des degrés de chaque nœud du graphe. La distance de similarité entre deux graphes est défini comme le nombre minimum des opérations primitives qui sont nécessaires pour que les deux graphes possèdent la même séquence de degrés.

Pour calculer la mesure de similarité, un histogramme trié de graphe est introduit dans [31]. Soit un graphe $G = (V, E)$, son histogramme est construit par le calcul des degrés de chaque nœud de G , par une suite où chaque nœud correspond à une case différente de l’histogramme. En triant ces cases, on obtient un histogramme trié de graphe G (sorted graph histogram). Les auteurs montrent que la L1-distance entre deux histogrammes triés de graphes est égal, à la distance de similarité entre les deux graphes correspondants. En plus, cette distance satisfait les trois propriétés fondamentales de l’espace métrique : la positivité, la symétrie et l’inégalité triangulaire.

Évidemment, si les graphes dans une base de données n’ont pas le même ordre les histogrammes triés des ces graphes auront des dimensions différentes. Pour permettre l’utilisation des structures d’indexation des

espaces de vecteurs, une technique de pliage pour les histogrammes a été introduite pour achever à une dimension constante des histogrammes de chaque graphe. Pour créer un histogramme plié à partir d'un histogramme trié, l'ordre maximum de tous les graphes dans la base ainsi que tous les graphes requêtes doit être connus à l'avance. Au début, la dimension désirée n est fixée, et pour les graphes avec un nombre de nœuds inférieur à n , la dimension de leurs histogrammes triés est ramenée à la dimension n . Dans tout les autres cas, un nouveau histogramme plié H_p sera construit par échantillonnage de la somme des n -premières valeurs de H_t . Après la résolution du problème de la dimension, les auteurs utilisent la technique d'indexation R-tree pour exploiter leur mesure.



La technique de pliage d'histogramme de Papadopoulos et Manolopoulos.

4.3 Mesure de similarité basée sur le plus grand sous-graphe commun

Cette méthode de mesure de similarité entre deux graphes proposée par Bunke et Shearer dans [10] est basée sur le plus grand sous-graphe commun de deux graphes. Un graphe G est appelé un sous-graphe commun aux deux graphes G_1 et G_2 , s'il est un sous graphe, respectivement, de G_1 et G_2 . Un sous-graphe commun G de deux graphes G_1 et G_2 est dit maximal (le plus grand) s'il n'existe aucun autre sous-graphe commun de G_1 et G_2 avec un ordre (taille, en terme de nombre de nœuds) plus grand que celui de G . Dans [10], le plus grand sous-graphe commun est noté " mcs " (*maximal common subgraph*), on note le plus grand sous-graphe commun de G_1 et G_2 par $pgsc(G_1, G_2)$. La mesure de similarité entre deux graphes non-vides G_1 et G_2 est définit par :

$$d_{pgsc}(G_1, G_2) = 1 - \frac{|pgsc(G_1, G_2)|}{\max\{|G_1|, |G_2|\}}$$

Où $|pgsc(G_1, G_2)|$ est le nombre des nœuds du $pgsc$ de G_1 et G_2 , et $|G_1|$ et $|G_2|$ sont respectivement le nombre de nœuds de G_1 et G_2 . Cette mesure selon [10] respecte les propriétés suivantes :

- $0 \leq d_{pgsc}(G_1, G_2) \leq 1$
- $d_{pgsc}(G_1, G_2) = 0 \leftrightarrow G_1$ et G_2 sont isomorphes l'un à l'autre.
- $d_{pgsc}(G_1, G_2) = d_{pgsc}(G_2, G_1)$
- $d_{pgsc}(G_1, G_3) \leq d_{pgsc}(G_1, G_2) + d_{pgsc}(G_2, G_3)$

4.4 La mesure de similarité ϕ -distance

Une autre mesure de similarité pour les graphes est proposée par Chartrand, Kubicki et Schultz dans [18]. Cette mesure est basée sur des fonctions entre les ensembles des nœuds des graphes, qui sont comparés, et est définit pour les graphes connexes de même ordre. La mesure de similarité se base sur la ϕ -distance définit par : Soient les deux graphes connexes $G_1 = (V_1, E_1)$ et $G_2 = (V_2, E_2)$ ayant le même ordre n , et l'application linéaire $\phi : V_1 \mapsto V_2$. La ϕ -distance entre G_1 et G_2 est définit par :

$$Dist_\phi(G_1, G_2) = \sum_{u,v} |lp(u,v) - lp(\phi u, \phi v)|$$

Où la somme est prise pour toutes les paires des nœuds u et v distincts $\binom{n}{2}$ dans G_1 , et $lp(u,v)$ est la longueur du plus court chemin entre u et v dans G_1 . La mesure de similarité ϕ -distance entre deux graphes connexes G_1 et G_2 avec le même ordre est :

$$D_\phi(G_1, G_2) = \min \{ Dist_\phi(G_1, G_2) \}$$

Le minimum est pris parmi toutes les applications ϕ possibles de V_1 à V_2 .

Chartrand et al. montrent que la mesure de similarité ϕ -distance respectent les propriétés des espaces métriques, mais ils ne proposent aucun algorithme pour calculer cette mesure. En revanche ils présentent un filtre de limite inférieur pour la similarité. Ce filtre est basé sur la notion de la distance totale d'un graphe connexe G d'ordre n , définit par :

$$td(G) = \sum lp(u,v)$$

Où la somme est prise pour tous les paires des nœuds u et v distincts $\binom{n}{2}$ de G . Chartrand et al. [18] montre que pour deux graphes connexes G_1 et G_2 avec le même ordre on a :

$$|td(G_1) - td(G_2)| \leq D_\phi(G_1, G_2)$$

Évidemment, le calcul de la distance totale est très coûteux en matière de temps parce qu'il faut considérer toutes les paires de nœuds. Ce calcul peut être fait à l'avance de sorte que la distance totale puisse être employée comme une valeur filtre pour le graphe. Néanmoins, le calcul de la mesure de similarité ϕ -distance exacte demeure une tâche complexe, puisque ils existent $n!$ applications linéaire possibles entre les ensembles de nœuds de deux graphes avec l'ordre n .

4.5 La notion de graphe Médian

Étant donné un ensemble d'éléments, le médian peut être un concept très utile pour avoir une représentation qui accumule une information globale de l'ensemble. Dans le domaine de l'analyse d'image par approches structurales, le graphe médian, introduit par Jiang et Bunke dans [23, 21, 8], a pour objectif l'extraction de l'information essentielle à partir d'un ensemble de graphes dans un seul prototype (i.e. modèle).

Soit S un ensemble de graphes, le médian est défini comme le graphe dérivé de S qui a la plus petite somme de distances à tout les graphes dans l'ensemble S . Si la condition que le médian appartient à l'ensemble des graphes est requise, le médian de l'ensemble est obtenu. Sinon, un graphe médian généralisant est obtenu.

Soit U l'ensemble contenant tout les graphes qui peuvent être construit à partir d'un ensemble $S = \{g_1, g_2, g_3, \dots, g_n\}$ (i.e. tout graphe avec un nombre de nœud entre 1 et la somme de tous les nœuds de g_i et avec les étiquettes des nœuds et des arcs prises par les même deux fonctions qui ont affecté les étiquettes dans S). Le graphe médian généralisant et le graphe médian sont respectivement définis par :

$$\begin{aligned} \bar{g} &= \arg \min_{g \in U} \sum_{i=1}^n d(g, g_i) \\ \hat{g} &= \arg \min_{g \in S} \sum_{i=1}^n d(g, g_i) \end{aligned}$$

Avec $d(\cdot, \cdot)$ une certaine fonction de distance qui mesure la dissimilarité entre deux graphes.

Le concept de graphe médian généralisant est plus intéressant parce qu'il a un grand potentiel de capter l'information essentiel de l'ensemble des graphes. Mais aussi plus complexe à calculer, en fait sa complexité grandit exponentiellement en fonction du nombre de graphes et de leurs tailles. Par contre, le graphe médian peut être calculé en $O(n^2)$ étapes (pour chaque étape, nous aurons besoin de calculer la distance entre deux graphes).

Même si le problème de graphe médian généralisant est quelque peu récent, la littérature dans le domaine de l'analyse d'image et de la reconnaissance de forme par approches structurales, présente quelques travaux qui proposent différentes méthodes pour calculer le graphe médian généralisant. Par exemple, Jiang [22] utilise un algorithme génétique avec représentation chromosomique. Hlaoui [20] propose une solution basée sur la décomposition de problème de la minimisation de la somme des distances en deux parties; une pour la somme des distances dédiée aux nœuds et l'autre pour les arcs. Encore, la théorie de graphes spectrale [17] ainsi que la programmation linéaire [27] ont été utilisées pour cette problématique.

5 Applications utilisant des graphes

Dans le domaine de l'analyse d'images, les graphes ont été largement utilisés dans différents sous-domaines. Dans cette partie, nous nous focalisons sur les exploitations des graphes les plus intéressants dans le domaine de la segmentation, la recherche par contenu et la reconnaissance de formes. Ces choix ne sont pas exhaustifs mais ils nous paraissent important et plus ou moins connexes.

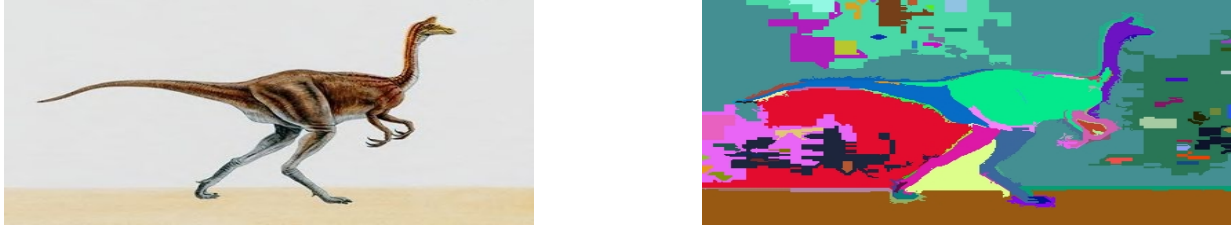


FIG. 2 – image segmentée par l'algorithme [16]

5.1 Segmentation

Soit $G = (V, E)$ un graphe non-orienté. Les nœuds $v_i \in V$ sont l'ensemble des éléments à segmenter et les arcs (v_i, v_j) correspondent aux paires des nœuds voisins. Chaque arc (v_i, v_j) à un poids correspondant $p((v_i, v_j))$, qui est une mesure de dissimilarité positive entre les éléments voisins v_i et v_j . Concrètement, les éléments dans V sont les pixels de l'image et le poids est une certaine mesure de dissimilarité entre deux pixels connectés par un arc (différence d'intensité, couleur, localisation ...). Dans les approches de segmentation basées sur les graphes, une segmentation S est une partition de V aux composants tels que chaque composant (ou région) $C \subseteq S$ correspond à un composant connexe dans un graphe $G' = (V, E')$, où $E' \geq E$. Autrement dit, chaque segmentation est induite par un sous-ensemble des arcs de E . Il existe plusieurs façons de mesurer la qualité d'une segmentation mais en général nous voulons que les éléments dans une région soient similaires, et les éléments dans des régions différentes soient dissimilaires. Cela veut dire que les arcs entre deux nœuds dans le même composant doivent avoir un poids faible, et les arcs entre les nœuds dans des composants différents doivent avoir un poids élevé. Différents algorithmes [16, 5] s'appuient sur cette représentation pour segmenter des images. La Fig. 2 montre un exemple de segmentation dans ce contexte. La figure montre bien que la segmentation est sensible au choix de la mesure entre les pixels p . Un choix de mesure de dissimilarité faible a tendance à produire une sur-segmentation de l'image alors qu'une mesure élevée à produire une sous-segmentation de l'image. Dans le cas de la figure 2 nous constatons une sur-segmentation du fond de l'image alors qu'à première vue cette zone peut nous paraître homogène.

5.2 Recherche par le contenu

Dans le contexte de la recherche par contenu d'images dans des bases de données, lorsque l'utilisateur formule une requête visuelle, sa cible de recherche est rarement représentée par une image entière comme le suppose le paradigme classique de recherche par une image exemple. L'image ne doit pas être traitée comme une unité atomique, car elle est généralement constituée d'un ensemble composite de zones visuelles exprimant une certaine sémantique. Dans ce contexte, les graphes, de par leurs natures, proposent une solution ajustée à ce problème. Bien évidemment, l'utilisation des graphes augmente radicalement la complexité (voir [12]) des tests d'évaluation de similarités entre les entités, mais les récents et rapides développements des techniques d'indexation multimédia ont proposé des solutions pour réduire la complexité d'appariement.

Berretti et al. [6] a proposé une technique d'appariement et d'indexation dédiée aux modèles de graphes dans la recherche par contenu. En fait, les images sont représentées par des graphes relationnels attribués "*Attributed relational graph (ARG)*" (image = $\langle \mathbf{E}, \mathbf{a}, \mathbf{w} \rangle$, où \mathbf{E} représente l'ensemble des nœuds, \mathbf{a} les attributs de nœuds et \mathbf{w} les attributs des arcs appelés "relations spatiales binaires"). Ensuite, Berretti propose un modèle de comparaison des "ARG" avec trois distances (distance d'attribut, distance de relation et distance commune). En adaptant la technique d'indexation m-tree [11], les auteurs [6] présentent un test sur la bibliothèque d'image web-museum (<http://www.oir.ucf.edu>). Dans [15, 19] d'autres solutions de recherche par composantes visuelles font recours aux graphes, notamment les graphes d'adjacence des régions "*Regions Adjacency Graph (RAG)*" également utilisé dans [26] pour la reconnaissance de symboles dessinés à la main dans des documents de plans architecturaux. En l'occurrence, d'autres types de graphes ont été élaborés dans les travaux sur la recherche par le contenu comme le graphe de variance spatiale et le graphe d'adjacence des couleurs modifiés proposés dans [32] l'auteur présente un test d'un algorithme de recherche par contenu d'images couleurs basé sur une représentation sous forme de graphe hybride construisant des

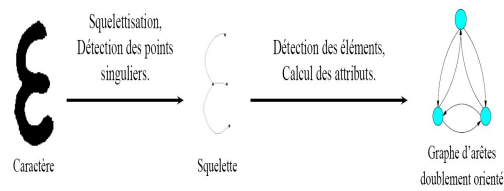


FIG. 3 – Représentation sous forme de graphe d'un caractère manuscrit [2]

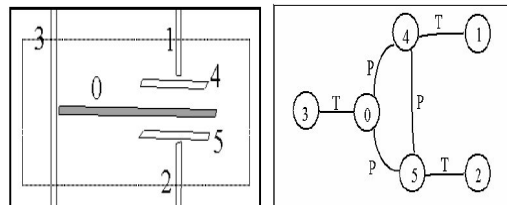


FIG. 4 – Représentation sous forme de graphe d'une forme [33]

caractéristiques pour une image selon l'adjacence des régions et la distribution des couleurs dans chaque régions.

Certainement, la représentation sous forme de graphes est une solution adéquate pour la recherche par contenu d'images. Cependant, la problématique principale est le choix des informations à stocker dans ces graphes, par conséquent, les informations sont sélectionnées selon le type d'images à traiter. Par ailleurs, la complexité des algorithmes d'appariement présente un handicap pour l'utilisation plus générale des graphes.

5.3 Reconnaissance de formes

Plusieurs travaux [8, 2, 14, 13, 33] ont montré l'utilité des graphes dans la reconnaissance de formes manuscrits ou non. Dans ce cas, la quantité d'information contenue dans le graphe offre un bon compromis entre la précision de la description et la tolérance aux déformations que l'on observe dans l'écriture manuscrite [13]. Les graphes ont été utilisés comme structure de représentation (Fig. 3 et 4) qui permet non seulement à l'appariement des formes mais aussi comme une structure intermédiaire suffisante capable de s'adapter pour décrire d'autres formes (manuscrit) [33]. Nous invitons le lecteur intéressé vers l'article de Horst Bunke [8] qui présente un état de l'art complet de l'utilisation des graphes dans la reconnaissance de formes.

6 Discussion et conclusions

Dans cet article nous avons passé en revue des applications de la théorie des graphes dans le domaine de l'analyse d'images notamment les mesures de similarité développées pour mesurer la similarité d'images représentées par des graphes. La mesure de similarité de deux images se mesure en appariant les nœuds des graphes associés aux images de façon à retrouver leurs caractéristiques communes. Différentes sortes d'appariements de graphes, impliquant différentes mesures de similarité ont été proposés dont nous avons cité une liste des travaux non-exhaustive. Nous pouvons conclure que les graphes sont une forme de représentation très polyvalente et flexible adaptée au domaine de l'analyse d'images.

Les mesures de similarité passées en revue dans cet article sont très générales. En fait, elles peuvent s'appliquer dans des différents contextes sans restrictions sur les étiquettes et les arcs des graphes.

Malgré la complexité élevée des algorithmes d'appariement des graphes, la représentation structurelle des images reste une piste de recherche très prometteuse. Les quelques applications présentées dans cet article montre l'intérêt des graphes dans le traitement d'image. La flexibilité et la faculté à s'adapter aux différents type des données ainsi que la possibilité de traiter une sous partie sans avoir besoin de considérer une image

comme un tout, toutes ces caractéristiques sont des atouts pour les approches basées sur les graphes dans le domaine de l'analyse d'images et la recherche d'informations notamment la recherche par le contenu d'images.

Références

- [1] R. Ambauen, S. Fisher, and B. H. Graph edit distance with node splitting and merging, and its application to diatom identification. *IAPR-TC15 Workshop on GbRPR, LNCS 2726*, pages 95–106, 2003.
- [2] D. Arrivault. *Apport des Graphes dans la Reconnaissance Non-Contrainte de Caractères Manuscrits Anciens*. PhD thesis, Université de Poitiers, 2006.
- [3] R. Baeza-Yates and G. Valiente. An image similarity measure based on graph matching. *Proc. International Symposium on String Processing Information Retrieval*, pages 28–38, 2000.
- [4] H. G. Barrow and R. J. Popplestone. Relational descriptions in picture processing. In *Machine Intelligence*, 4 :377–396, 1971.
- [5] F. P. Bergo, A. X. Falcao, P. A. Miranda, and L. M. Rocha. Automatic image segmentation by tree pruning. *Math Imaging Vision*, 29 :141–162, 2007.
- [6] S. Berretti, A. D. Bimbo, and E. Vicario. Efficient matching and indexing of graph models in content-based retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10) :1089–1105, 2001.
- [7] D. Blostein. Graph transformation in document image analysis : Approaches and challenges. *GbRPR 2005, LNCS 3434*, pages 23–34.
- [8] H. Bunke. Recent advances in structural pattern recognition with applications to visual form analysis. *Visual Form 2001, Springer Verlag, LNCS 2059*.
- [9] H. Bunke. Attributed of programmed graph grammars and their application to schematic diagram interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(6) :574–582, Novembre 1982.
- [10] H. Bunke and K. Shearer. A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters*, 19(3-4) :255–259, march 1998.
- [11] P. Ciaccia, M. Patella, and P. Zezula. M-tree : An Efficient Access Method for Similarity Search in Metric Spaces. In *Proc. of the 23rd Int. Conf. Very Large Database Conference (VLDB'97)*, Athens Greece, 1997.
- [12] D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(3) :265–298, 2004.
- [13] P. Dargentou. *Contribution à la segmentation et à la reconnaissance de l'écriture manuscrite par l'ordinateur*. PhD thesis, INSA LYON, 1994.
- [14] C. Di Ruberto, G. Rodriguez, and L. Casta. Recognition of shapes by morphological attributed relational graphs. *AIIA*, 2002.
- [15] J. Fauqueur. *Contributions pour la Recherche d'Images par Composantes Visuelles*. PhD thesis, INRIA - Université Versailles St Quentin, 2003.
- [16] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2), Septembre 2004.
- [17] M. Ferrer, F. Serratosa, and A. Sanfeliu. Synthesis of median spectral graph. *IbPRIA, LNCS 3523*.
- [18] G. Chartrand, G. Kubicki, and M. Schultz. Graph similarity and distance in graphs. *Aequationes Mathematicae*, 55(1-2) :129–145, february 1998.
- [19] H. Grecu and P. Lambert. Représentation floue et graphe d'adjacence pour la simplification d'images segmentées : Application à l'indexation. *10ème colloque GRETSI, Vannes*, 1999.
- [20] A. Hlaoui and S. Wang. A new median graph algorithm. *IAPR Workshop on GbRPR, LNCS 2726*, pages 225–234, 2003.
- [21] X. Jiang and H. Bunke. Optimal lower bound for generalized median problems in metric space. *IAPR International Workshop on Structural, Syntactic and Statistical Pattern Recognition (S+SSPR), Windsor, Canada, LNCS 2396*, pages 143–151, 2002.
- [22] X. Jiang, A. Munger, and H. Bunke. Scomputing the generalized median of a set of graphs. *2nd IAPR-TC-IS Workshop on Graph Based Representations*.
- [23] X. Jiang, A. Munger, and H. Bunke. On median graphs :properties, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10) :1144–1151, October 2001.
- [24] A. Karray. Recherche de lettres par le contenu. Master's thesis, Laboratoire L31, Universités de La Rochelle et de Sfax, France et Tunisie, 2006.
- [25] C.-Y. Li and C.-T. Hsu. Region correspondence for image retrieval using graph-theoretic approach and maximum likelihood estimation. *International Conference on Image Processing*, pages 421–424, 2004.
- [26] J. Lladòs, E. Martí, and J. J. Villanueva. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10) :1137–1143, 2001.
- [27] L. Mukherjee, V. Singh, J. Peng, J. Xu, M. Zeitz, and R. Berezney. Generalized median graphs : Theory and applications.
- [28] M. Neuhaus and H. Bunke. A probabilistic approach to learning costs for graph edit distance. *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, 3 :389–393, 2004.
- [29] I. Ounis. *Un modèle d'indexation relationnel pour les graphes conceptuels fondé sur une interprétation logique*. PhD thesis, Grenoble, 1998.

- [30] J. Pan, H. Yang, C. Faloutsos, and P. Duygulu. Gcap : Graph-based automatic image captioning. *In Proceedings of the 4th International Workshop on Multimedia Data and Document Engineering*, 2004.
- [31] A. N. Papadopoulos and Y. Manolopoulos. Structure-based similarity search with graph histograms. *Proceedings of International Workshop on Similarity Search (DEXA IWOSS'99)*, pages 174–178, Septembre 1999.
- [32] I. Park, I. Yun, and S. Lee. color image retrieval using hybrid graph representation. *Image and Vision Computing*, 17 :465–474, 99.
- [33] J. Ramel, N. Vincent, and H. Emptoz. A structural representation for understanding line-drawing images. *IJDAR*, 3 :58–66, 2000.
- [34] A. Robles-Kelly and E. Hancock. Graph edit distance from spectral seriation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3) :365–378, 2005.
- [35] R. A. Wagner and M. J. Fisher. The string-to-string correction problem. *Journal of the ACM*, 21(1) :168–173, 1974.
- [36] L. Wiskott, J. Fellous, N. Kruger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7) :775–779, July 1997.