

Classification de formes avec un noyau sur graphes flexible et robuste au bruit

François-Xavier Dupé¹

Luc Brun¹

¹ GREYC, UMR CNRS 6072 - ENSICAEN - Université de Caen Basse-Normandie,
6 boulevard du Maréchal Juin,
14050 Caen France,
francois-xavier.dupe, luc.brun@greyc.ensicaen.fr

Résumé

La squelettisation par axe médian étant une transformation homotopique, le squelette d'une forme 2D correspond à un graphe planaire dont les faces codent les trous et les sommets chaque jonction et extrémité. Ce graphe n'est pas un graphe simple, car composé de boucles internes et d'arêtes multiples à cause des trous. Dans le cadre de la comparaison de formes, celui-ci est souvent transformé en une structure plus simple comme un arbre ou un graphe simple, perdant de ce fait des informations importantes sur la forme. Dans ce papier, nous proposons un noyau sur graphes combinant un noyau sur sacs de chemins et un noyau sur faces. Les chemins sont définis à partir du graphe non simple et le noyau sur chemins est renforcé par un processus d'édition. Le noyau sur faces reflète l'importance des trous dans une forme, cette information pouvant être une caractéristique importante de la forme. Le noyau résultant est un noyau défini positif, compétitif avec les noyaux proposés dans l'état de l'art.

Mots Clef

Reconnaissance de formes, Squelette, Machines à noyaux, Noyaux sur graphes

Abstract

The medial axis being an homotopic transformation, the skeleton of a 2D shape corresponds to a planar graph having one face for each hole of the shape and one node for each junction or extremity of the branches. This graph is non simple since it can be composed of loops and multiple-edges. Within the shape comparison framework, such a graph is usually transformed into a simpler structure such as a tree or a simple graph hereby losing major information about the shape. In this paper, we propose a graph kernel combining a kernel between bags of trails and a kernel between faces. The trails are defined within the original complex graph and the kernel between trails is enforced by an edition process. The kernel between bags of faces allows to put an emphasis on the holes of the shapes and hence on their genre. The resulting graph kernel is positive definite and competitive with state of art kernels.

Keywords

Shape Recognition, Skeleton, Kernel Machine, Graph Kernel

1 Introduction

La squelettisation à partir de l'axe médian est une transformation homotopique. Le squelette d'une forme 2D est une structure linéaire 2D avec autant de trous que la forme. Une manière naturelle d'encoder une telle structure est d'utiliser un graphe en créant un sommet pour chaque jonction et extrémité et une arête pour chaque branche entre deux sommets. Ce graphe est un graphe planaire non simple qui peut être enrichi avec des informations comme l'évolution du rayon du cercle maximal le long des branches [22, 19, 23, 1, 7]. Le problème de comparaison de forme est ainsi transformé en un problème de comparaison de graphes. Cependant, les méthodes de comparaisons de graphes robustes au bruit structurel, comme la méthode du plus grand sous-graphe commun ou le problème équivalent de distance d'édition entre graphes [3] ont en général une complexité exponentielle. Beaucoup d'auteurs utilisent donc un encodage plus simple du squelette conduisant à une fonction de comparaison avec une complexité réduite. Siddiqi [22] et Sebastian [20] transforment les graphes en arbres puis appliquent un schéma de comparaison d'arbres. Une autre méthode, introduite par Pelillo [18], transforme les graphes en arbres puis modélise le problème de comparaison d'arbres en un problème de recherche de cliques maximales dans un graphe d'association spécifique. Une dernière méthode proposée par Bai et Latecki [1] alignent les extrémités (sommets de degré un) puis comparent les chemins les plus courts entre chaque paire de sommets. Contrairement aux approches précédentes, cette dernière méthode fonctionne avec les boucles présentes dans un graphe complexe et donc prend les trous en compte. Les méthodes à noyaux sur graphes permettent d'utiliser un grand nombre de méthodes statistiques (SVM, PCA, ...) sur des ensembles de graphes. Ces méthodes nécessitent la définition d'une fonction de similarité entre deux graphes appelé noyau. Neuhaus et Bunke [17] proposent un élégant schéma de construction de noyaux sur graphes basé sur les distances d'éditions. Cependant, ces noyaux ne sont

généralement pas définis positifs. Notons que le fait de ne pas être défini positif n'est pas rédhibitoire puisque les résultats d'un SVM peuvent être interprétés lorsque le noyau n'est pas défini positif [10] et que certains classificateurs ont été conçus spécifiquement pour de tels noyaux [9]. Cependant, l'utilisation de noyaux non définis positifs complexifie l'interprétation des résultats et affaiblit généralement les propriétés vérifiées par ceux-ci. Par exemple, dans le cas du SVM, la solution trouvée par celui-ci n'est plus un optimum global mais un optimum local. Une autre solution consiste à utiliser des noyaux sur graphes comme le noyau sur marches aléatoires ou le noyau sur graphes marginalisé [26] qui sont définis positifs. Toutefois, bien que ces noyaux soient plus simple d'utilisation, ils n'ont pas la robustesse au bruit fournies par les noyaux basés sur les distances d'édits.

Ce papier suit une première contribution [5] où nous avons défini un schéma de construction de noyaux sur graphes qui peut intégrer une notion de réécriture afin d'être robuste au bruit. Cependant, cette méthode est définie pour des arbres et donc ne prend pas correctement en compte les trous dans les formes. Nous présentons tout d'abord la construction des graphes (Section 2) à partir des formes. Puis, nous rappelons quelques définitions et étendons notre schéma de construction aux parcours élémentaires (Section 3). Nous proposons ensuite d'étendre le processus de réécriture, initialement défini pour des arbres, aux graphes (Section 4). Puis, nous proposons de combiner notre noyau sur graphes avec un noyau sur faces qui compare les faces des graphes planaires (Section 5). Enfin, une expérience avec un classifieur multi-classes est proposée pour illustrer l'importance de la prise en compte des trous (Section 6).

2 Graphes basés sur le squelette

Le squelette d'une forme est généralement construit à partir de l'axe médian qui est défini comme les centres des cercles de rayon maximal [21]. Beaucoup de graphes peuvent être associés au squelette d'une forme. Notre schéma de construction de graphe suit l'approche proposée par Siddiqi [22]. Il considère la traduction enrichie du squelette d'une forme en une structure de graphe :

- Les sommets correspondent aux points terminaux, aux jonctions et à tous les points codant une importante accélération ou décélération de l'évolution du rayon le long d'une branche. Les pentes décrivant la vitesse de changement du rayon le long d'une branche peuvent être obtenues avec des méthodes de régression basées sur des splines de premier ordre [5, 4].
- Les arêtes correspondent aux branches du squelette entre deux sommets.

La figure 1 montre trois formes avec leurs points d'inflexions. La détection de ces points peut varier selon les paramètres de la méthode de régression utilisée. Une telle variabilité doit être compensée par la flexibilité de notre noyau sur graphe.

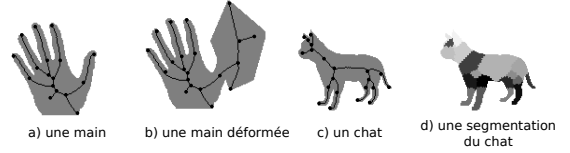


FIG. 1: Trois exemples de squelettes avec leurs points d'inflexion.

La graphe associé à la forme propose uniquement des informations topologiques. Des propriétés géométriques additionnelles doivent donc être ajoutées aux sommets et aux arêtes. D'un point de vue structurel, un sommet représente un point particulier dans le squelette et une arête une branche. Cependant, une branche représente aussi un ensemble de points qui sont plus près de cette branche que des autres. Cet ensemble de points est défini comme la *zone d'influence* de la branche et peut être calculé en utilisant une transformation de type SKIZ [16] (Fig. 1d).

Nous avons sélectionné un ensemble d'attributs qui produisent une bonne approximation de la description de la forme. Torsello [25] propose comme attribut d'arête une approximation du périmètre de la bordure ayant contribué à la formation de la branche représentée par l'arête. Cette caractéristique présente le double avantage d'être additive et de coder l'importance des arêtes au sein d'un graphe. Suard [24] propose comme attribut de sommet, la distance entre le point représenté par le sommet et le centre de gravité de la forme. Cette distance est normalisée en la divisant par la racine carrée de la surface de la forme.

Deux descripteurs de branches sont aussi proposés : l'évolution du rayon du cercle maximal et de la tangente le long de la branche représentées. Pour chaque point $(x(t), y(t))$ d'une branche, $t \in [0, 1]$, nous considérons le rayon $R(t)$ du cercle maximal. De manière à normaliser les données, le rayon est divisé par la racine carrée de l'aire de la zone d'influence de la branche. Nous introduisons aussi $\alpha(t)$, l'angle formé par le vecteur tangente en $(x(t), y(t))$ et l'axe des abscisses. Ainsi, les deux descripteurs considérés sont les coefficients $(a_k)_{k \in \mathbb{N}}$ et $(b_k)_{k \in \mathbb{N}}$ des polynômes de régressions sur $R(t)$ et $\alpha(t)$ au sens des moindres carrés. Si les deux polynômes sont d'un ordre suffisant, le squelette peut être reconstruit à partir du graphe et donc la forme aussi. Une analyse plus détaillée de ce schéma de construction du graphe peut être trouvé dans [5].

3 Noyaux sur sacs de chemins

Considérons un *graphe* $G = (V, E)$, avec V l'ensemble des sommets et $E \subseteq V \times V$ l'ensemble des arêtes. Nous définissons un *graphe simple* comme un graphe sans arêtes multiples et sans boucles (une arête reliant un sommet à lui-même). Nous définissons un *chemin* comme une suite alternée de sommets et d'arêtes avec des arêtes distinctes et un *chemin simple* comme un chemin avec des sommets distincts. Un *chemin fermé* est un chemin dont le premier

sommet coïncide avec le sommet final. Un *sac de chemins* T associé à un graphe G est défini comme un ensemble de chemins dans G dont la cardinalité est notée $|T|$. Enfin, nous notons par K_{chemin} un noyau générique sur chemins.

3.1 Noyaux moyennes

En considérant les sacs comme des ensembles, Suard [23] a proposé plusieurs noyaux sur sacs de chemins simples extensibles directement aux chemins. Parmi ces noyaux, le noyau moyenne est proposé comme un noyau de convolution [11] entre chemins : soient T_1 et T_2 deux sacs de chemins, le *noyau moyenne* entre deux sacs est défini par :

$$K_{moyenne}(T_1, T_2) = \sum_{t \in T_1} \sum_{t' \in T_2} p(t|T_1)p(t'|T_2) K_{chemin}(t, t'). \quad (1)$$

avec $p(t|T)$ la probabilité de t sachant T . Ces probabilités peuvent être définies par l'intermédiaire de lois marginales comme proposées par Kashima [14]. Ce noyau est défini positif si et seulement si K_{chemin} est un noyau défini positif.

Le principal défaut de ce noyau est que l'information est diluée lorsque les sacs sont composés de nombreux chemins. Une telle perte d'information peut être évitée en utilisant un noyau moyenne pondérée [6]. Ce noyau suppose que les chemins les plus pertinents sont proches du chemin moyen. Soient T_1 et T_2 deux sacs de chemins, le *noyau moyenne pondérée* est défini par :

$$K_{pondere}(T_1, T_2) = \frac{1}{|T_1|} \frac{1}{|T_2|} \sum_{t \in T_1} \sum_{t' \in T_2} \langle K_{chemin}(t, m), K_{chemin}(t', m') \rangle^d \frac{\omega(t)}{W} \frac{\omega(t')}{W'} K_{chemin}(t, t'). \quad (2)$$

où $d \in \mathbb{R}^+$, m et m' sont les chemins moyens de T_1 et T_2 , $\omega(t)$ (resp. $\omega(t')$) est le poids cumulé des arêtes de t (resp. de t') et W (resp. W') le poids total du graphe contenant t (resp. t'). Le noyau sur chemins entre un chemin t et le chemin moyen m est défini par : $K_{chemin}(t, m) = \frac{1}{|T|} \sum_{t_i \in T} K_{chemin}(t, t_i)$. Le noyau moyenne pondérée est un noyau de convolution composé d'un produit scalaire (similarité avec le chemin moyen) et le noyau sur chemin K_{chemin} . Il est donc défini positif si et seulement si K_{chemin} l'est.

3.2 Un premier noyau sur chemin

Pour son noyau marginalisé, Kashima propose un noyau sur chemin basé sur un produit de tenseur [14]. Son noyau est disponible pour toutes les définitions de chemins dans un graphe. Soient t et t' deux chemins, le *noyau sur chemin* noté $K_{classique}$ est défini comme valant 0 si $|t| \neq |t'|$ et comme suit sinon :

$$K_{classique}(t, t') = K_v(\varphi(v_1), \varphi(v'_1)) \prod_{i=2}^{|t|} K_e(\psi(e_{v_{i-1}v_i}), \psi(e_{v'_{i-1}v'_i})) K_v(\varphi(v_i), \varphi(v'_i)), \quad (3)$$

où $\varphi(v)$ et $\psi(e)$ sont respectivement les vecteurs de caractéristiques associées au sommet v et à l'arête e . K_v et K_e sont deux noyaux sur respectivement les sommets et les arêtes. $K_{classique}$ est un noyau construit par produit de tenseur. Il est donc défini positif si et seulement si K_v et K_e sont deux noyaux définis positifs. Dans cet article, nous utilisons des noyaux gaussiens basés sur des distances entre attributs.

4 Noyau d'édition sur chemin

Le principal problème avec les graphes basés sur les squelettes est que deux graphes différents peuvent représenter deux formes similaires. Deux types différents de bruit structurel peuvent apparaître dans un squelette : les ligatures produites par un bruit le long de la bordure et les élongations produites par une déformation générale de la forme. Généralement, ce bruit est géré par des opérations d'édition sur les graphes. Cependant, dans un sac de chemins nous devons considérer des opérations d'édition sur des chemins et non des graphes. L'effet du bruit structurel sur les chemins est double : ajout d'arêtes et de sommets.

4.1 Zone d'influence d'un sommet

Nous supposons que les arêtes de notre graphe sont associées à un poids qui code leur importance. Avec ce poids, nous calculons l'importance de chaque sommet et arête dans un chemin : l'importance d'une arête correspond à son poids. L'importance d'un sommet correspond au poids du sous-graphe (i.e. la somme des poids de ses arêtes) connecté au chemin par ce sommet. Lorsque les graphes sont des arbres [5], les sous-graphes sont des sous-arbres et le calcul de l'importance du sommet est sans ambiguïté. Par exemple, Fig. 2a montre un chemin au sein d'un arbre où les sous-arbres associés à chaque sommet sont clairement définis ainsi que leur poids.

Cependant, avec les formes trouées, les graphes ne sont pas des arbres et la définition de l'importance d'un sommet n'est pas directe. En effet, les sous-graphes peuvent être connectés à plusieurs sommets d'un chemin. Nous proposons de résoudre cette difficulté en utilisant l'algorithme de diffusion par marches aléatoires proposé par Grady [8], les poids normalisés des arêtes étant considérés comme des probabilités de transition. Pour chaque sommet v_i d'un chemin, l'algorithme de diffusion associe à chaque sommet v_l du graphe la probabilité $p_{l,i}$ qu'un marcheur aléatoire partant de v_l rejoigne v_i . Chaque sommet du graphe est associé au sommet du chemin avec lequel la probabilité est maximale. Le sous-graphe induit par l'ensemble des sommets associé à un sommet du chemin est désigné comme la zone d'influence de ce sommet. Cependant, cet algorithme est conçu pour des graphes simples. Nous effectuons donc les simplifications suivantes lors du calcul des marches aléatoires : les boucles sont supprimées et les arêtes multiples sont fusionnées en une simple arête dont le poids est la somme des poids des arêtes fusionnées.

Le poids d'une zone d'influence du sommet v est défini

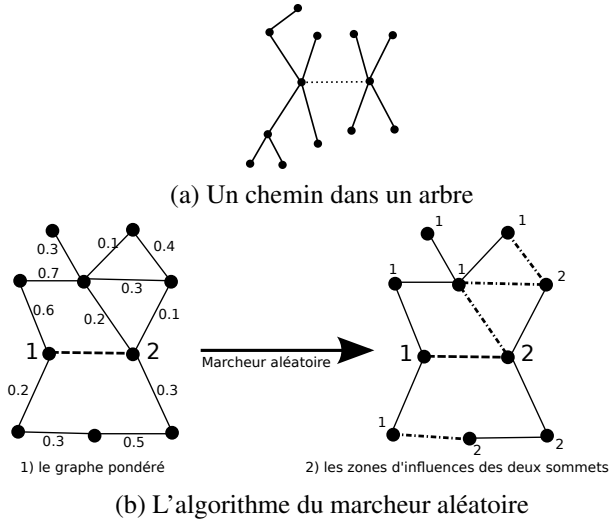


FIG. 2: Zones d'influence : (a) Exemple de chemin (arêtes hachurées) dans un arbre. (b) L'influence des sommets 1 et 2 d'un chemin (arêtes hachurées) dans un graphe calculée avec le marcheur aléatoire.

comme la somme 1) du poids des arêtes à l'intérieur de la zone et 2) d'une fraction du poids des arêtes partagées avec d'autres zones (i.e. les arêtes dont les sommets appartiennent à deux zones différentes). Par exemple, les arêtes hachurées et pointées dans Fig. 2b sont partagées par deux zones. Soient v_1 et v_2 deux sommets incidents d'une arête de poids w , v_1 (resp. v_2) est associé à sa zone d'influence avec une probabilité p_1 (resp. p_2). Nous définissons alors par $\frac{p_1}{p_1+p_2}w$ (resp. $\frac{p_2}{p_1+p_2}w$) la fraction du poids revenant à la zone d'influence possédant v_1 (resp. v_2). Fig 2b montre un exemple de zones d'influences (le chemin est défini par les arêtes hachurées) : remarquez l'importance de la zone associée au sommet 1 par rapport à celle associée au sommet 2.

4.2 Opérations élémentaires d'édition sur les chemins

L'opération de *suppression de sommet* supprime un sommet d'un chemin ainsi que la sous-structure du graphe (zone d'influence) reliée au chemin par le sommet. Dans le chemin, les deux arêtes incidentes au sommet sont fusionnées. Cette opération correspond à la suppression d'une partie de la forme. Par exemple, si nous enlevons le sommet 2 dans Fig. 3a, une nouvelle forme similaire à Fig. 3b est obtenue.

L'opération de *contraction d'arête* contracte une arête puis fusionne les deux sommets aux extrémités. Cela conduit à une contraction de la forme. Par exemple, si nous contractons l'arête $e_{1,2}$ de la forme dans Fig. 3a nous obtenons une nouvelle forme qui est similaire à Fig. 3c.

Comme chaque opération est interprétée en tant que transformation de la forme, les descripteurs globaux doivent être mis à jour. De ce point de vue, notre méthode peut être

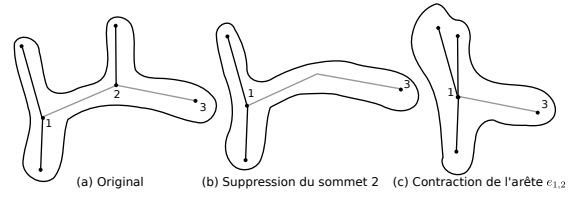


FIG. 3: Effet de l'édition sur la forme (chemin en gris)

considérée comme une combinaison des méthodes proposées par Sebastian [20] et Goh [7]. L'un utilise des descripteurs locaux avec des opérations d'édition et l'autre combine descripteurs locaux et globaux mais sans opérations d'édition pour comparer les formes.

4.3 Coût d'édition

Afin de guider le processus d'édition de chemins nous devons associer un coût à chaque opération d'édition élémentaire (Section 4.2). Considérons un attribut *poids* associé à chaque arête du graphe et codant la pertinence de la branche associée. Nous supposons que cet attribut est additif : le poids de deux arêtes consécutives le long d'un chemin est la somme des poids de ces deux arêtes.

Considérons maintenant un chemin $h = (v_1, \dots, v_n)$ dans un graphe G . Les coûts d'éditions associés à chacune des opérations dans h est défini par :

- Considérons un sommet v_i , $i \in \{2, \dots, n-1\}$ du chemin h (les sommets extrémités ne sont pas considérés). Le coût de l'opération de *suppression de sommet* sur v_i doit refléter deux de ses propriétés : 1) l'importance de la zone d'influence de ce sommet dans le graphe (Section 4.1) et 2) l'importance du changement de pente entre les deux arêtes dans le chemin $e_{v_{i-1}v_i}$ et $e_{v_iv_{i+1}}$ reliées par le sommet. L'importance de la zone d'influence est représentée par son poids total (Section 4.1) noté $\omega(v_i)$.

Nous représentons l'importance du changement de pente par l'angle $\beta(v_i)$ formé par les vecteurs de pente associés aux arêtes $e_{v_{i-1}v_i}$ et $e_{v_iv_{i+1}}$. Une valeur élevée code un changement important et inversement. Comme les pentes sont approchées par des polynômes de premier ordre (Section 2), l'angle $\beta(v_i)$ est donné par :

$$\beta(v_i) = \arccos \left(\frac{1 + a_1 a'_1}{\sqrt{1 + a_1^2} \sqrt{1 + a'^2_1}} \right) \quad (4)$$

avec a_1 et a'_1 les coefficients de pentes des polynômes.

Enfin le coût d'édition de suppression d'un sommet est défini par $(1 - \gamma)\omega(v_i) + \gamma 2\beta(v_i)/\pi$, où γ est une variable de réglage.

- Le coût de l'opération de *contraction d'arête* représente l'importance de l'arête dans le graphe G , ce qui est le propos du poids. Donc, le coût d'édition de contraction d'une arête est définie comme étant son poids.

Concernant le *poids* n'importe quelle mesure codant l'importance des branches d'un squelette peut être utilisé.

Nous avons choisi le périmètre normalisé proposé par Torsello [25] (Section 2), à cause de sa résistance au bruit sur la bordure de la forme.

4.4 Le noyau édition

Les fonctions de coût définies dans la section 4.3 nous permettent de considérer la fonction d'édition κ qui prend en paramètre un chemin et y applique l'opération la moins onéreuse. $\kappa^i(t)$ définit alors le chemin t après i éditions successives et nous notons $\text{cout}_i(t)$ le coût cumulé des i opérations d'éditions. Enfin, le noyau d'édition est défini comme un noyau de convolution entre les chemins et leurs réécritures :

$$K_{edit}(t, t') = \frac{1}{D+1} \sum_{k=0}^D \sum_{l=0}^D \exp\left(-\frac{\text{cout}_k(t) + \text{cout}_l(t')}{2\sigma_{cout}^2}\right) K_{classique}(\kappa^k(t), \kappa^l(t')), \quad (5)$$

où D est le nombre maximal d'édition et σ_{cout} est le paramètre du noyau gaussien pénalisant les comparaisons selon les coûts d'éditions. Ce noyau est un noyau de convolution [11]. Il est donc défini positif si et seulement si K_{chemin} est défini positif. Les expériences montrant le comportement de K_{edit} face au bruit structurel sont fournies dans [5].

5 Noyau sur chemin fermé

Les faces d'un graphe basé sur le squelette codent les trous présents dans la forme. Ils représentent donc une information importante sur celle-ci. Lorsque nous utilisons les noyaux précédents, les faces sont représentées par des chemins comme les autres. Donc lors de la construction des sacs, ces chemins particuliers peuvent ne pas apparaître dans le sac ou être noyé par le nombre. Ainsi, il est pertinent de mettre ces chemins en avant lorsque nous comparons des formes trouées.

Plusieurs noyaux sur les cycles ont été proposés pour les graphes [13]. Cependant ces noyaux ne sont pas conçus pour la classification de formes pour deux raisons : ils ne considèrent pas l'orientation des faces et ne sont pas restreints aux seuls cycles codant les faces.

Une comparaison efficace des faces pour la classification de formes demande un noyau robuste au bruit structurel. Nous proposons de coder chaque trou par un unique chemin fermé qui décrit la face correspondante. Ce chemin commence par le sommet le plus proche du centre de gravité et parcourt les arêtes dans le sens trigonométrique. Par exemple, le trou du cadenas (Fig. 4b) est décrit par le chemin fermé "4 e3 3 e1 4". Enfin, deux chemins fermés codant des faces sont simplement comparés en utilisant un noyau sur chemin tel que $K_{classique}$ (section 3.2) ou K_{edit} (section 4).

Cependant, lors de la comparaisons de chemins fermés, nous pouvons faire face à des erreurs d'alignements provoquer par une mauvaise sélection du sommet initial. Pour

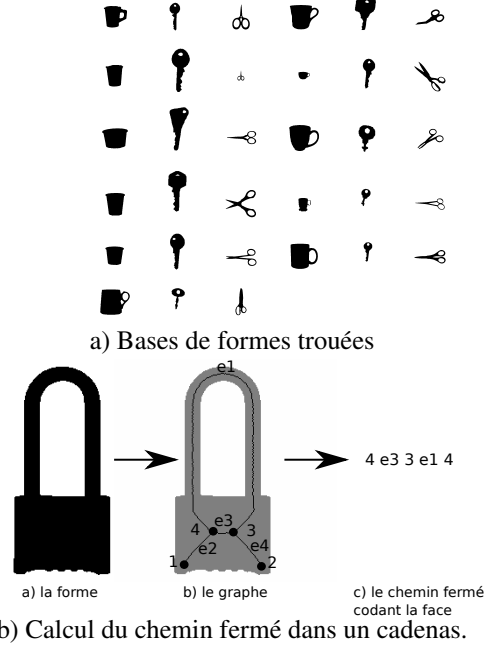


FIG. 4: Formes trouées et calcul de chemin fermé.

renforcer la robustesse de notre noyau, des versions décalées des chemins sont également comparées. Par exemple, la face dans Fig. 4b présente deux sommets à égale distance du centre de gravité ; le chemin fermé "3 e1 4 e3 3" est ainsi un chemin acceptable correspondant à une version décalée du chemin précédent. Nous définissons la fonction $\mu_i(t)$ qui applique un décalage circulaire de i arêtes du chemin t dans le sens horaire si i est positif et trigonométrique si i est négatif.

Le *noyau décalage* est donc défini comme la convolution pondérée entre les chemins et leurs versions décalées en utilisant un noyau sur chemins K_{chemin} :

$$K_{decalage}(t, t') = \frac{1}{(2p+1)^2} \sum_{i=-p}^p \sum_{j=-p}^p \exp\left(-\frac{|i|+|j|}{2\sigma_{ferme}^2}\right) K_{chemin}(\mu_i(t), \mu_j(t')), \quad (6)$$

où p est le nombre maximal de décalages. Ce noyau est défini positif si et seulement si K_{chemin} l'est. Enfin, le *noyau sur chemins fermés* est défini comme le noyau moyenne entre tous les chemins fermés codant les faces de deux graphes planaires G_1 et G_2 :

$$K_{chemin_ferme}(G_1, G_2) = \frac{1}{|C(G_1)|} \frac{1}{|C(G_2)|} \sum_{t \in C(G_1)} \sum_{t' \in C(G_2)} K_{decalage}(t, t'), \quad (7)$$

où $C(G_1)$ (resp. $C(G_2)$) représente l'ensemble des chemins fermés codant les faces dans G_1 (resp. G_2) et $|C(G_1)|$ (resp. $|C(G_2)|$) la cardinalité de l'ensemble $C(G_1)$ (resp.

$C(G_2)$). Ce noyau est défini positif si et seulement si K_{decale} l'est.

Au final, un noyau noté $K_{combine}$ est construit à partir des deux noyaux proposés :

$$K_{combine}(G_1, G_2) = (1 - \gamma)K_{pondere}(T_1, T_2) + \gamma K_{chemin_ferme}(G_1, G_2), \quad (8)$$

où T_1 (resp. T_2) est le sac de chemins associé à G_1 (resp. G_2), $\gamma \in [0, 1]$ est une variable de réglage, $K_{pondere}$ (2) est le noyau sur les sacs de chemins et K_{chemin_ferme} le noyau sur les chemins fermés (7). Ce noyau est défini positif sur l'union des domaines des sacs de chemins et des sacs de faces, car il définit comme l'addition de deux noyaux définis positifs multipliés par des coefficients positifs [2].

6 Expériences

Nous proposons deux expériences pour montrer les comportements de nos noyaux par rapport à l'état de l'art sur des formes trouées. Pour ces deux expériences, les sacs de chemins (Section 3) sont composés de 2 pourcent des chemins les plus pertinents parmi tous les chemins d'une taille allant jusqu'à 9 arêtes. Le nombre maximal d'éditations (Section 4) est fixé à 9 et le nombre maximal de décalages (Section 5) à 5. Les paramètres des différents noyaux gaussiens impliqués ont été réglés à la main après une première estimation avec une validation croisée. Pour des raisons de performances, le noyau sur les marches aléatoires [26] est utilisé sur une version augmentée des arbres de recouvrement maximal : lorsqu'une arête est impliquée dans la formation d'un cycle ou d'une boucle, nous changeons l'un des sommets incidents en un nouveau sommet de degré 1 avec les mêmes caractéristiques de manière à briser la boucle ou le cycle. Avec cette astuce, les graphes peuvent être codés par une matrice d'adjacence et nous pouvons profiter des méthodes de calculs rapide du noyau sur marches aléatoires [26]. Notons qu'une alternative existe consistant à utiliser les graphes d'arêtes (line graph).

6.1 Première expérience

La première expérience consiste en une classification une classe avec la méthode proposée par Hoffmann [12]. Pour les deux classes clés et ciseaux, nous prenons arbitrairement 3 formes pour former les ensembles d'entraînement. Puis, nous appliquons l'algorithme de classification. La figure 5 montre les courbes ROC correspondant aux résultats. Le noyau $K_{combine}$ présente les meilleurs résultats pour chacune des classes. Mettre une emphase sur les trous améliore donc les performances de classification. Cette remarque est confirmée par les résultats du noyau $K_{pondere}$. Le noyau sur les marches aléatoires présente les pires résultats, sauf pour la classe ciseaux où il est compétitif avec le noyau $K_{pondere}$.

6.2 Deuxième expérience

Nous proposons une deuxième expérience utilisant un classifieur multi-classes [27]. La base de formes est construite

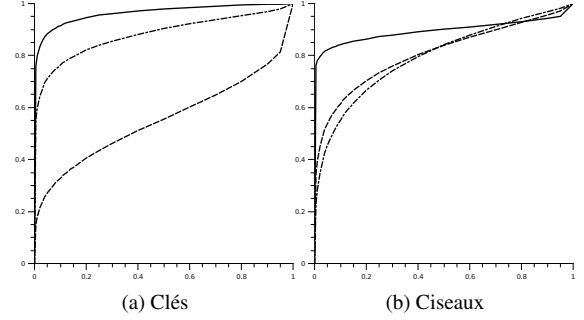


FIG. 5: Courbes ROC pour deux formes trouées en utilisant $K_{combine}$ (—), $K_{pondere}$ (---) et le noyau sur les marches aléatoires (-·-).

en ajoutant des formes avec des trous (Fig. 4a) à la base de Kimia avec 99 formes [15]. Trois noyaux sont testés : le noyau combinant le noyau moyenne pondérée avec le noyau sur les chemins fermés noté $K_{combine}$ (8), le noyau moyenne pondéré seul noté $K_{pondere}$ et le noyau sur les marches aléatoires [26]. Le noyau sur chemins utilisé avec les deux premiers noyaux est le noyau d'édition K_{edit} (Section 4). De plus, l'apport du noyau sur faces K_{chemin_ferme} lors de cette expérience peut être évalué en comparant les performances de $K_{combine}$ avec celles de $K_{pondere}$ puisque ce dernier noyau est égal à $K_{combine}$ dans le cas $\gamma = 0$ (8).

L'expérience consiste à classer d'une base complète en 5 classes (2 classes provenant de la base de Kimia et 3 classes de formes trouées). L'ensemble d'entraînement était constitué arbitrairement de 5 formes de chacune des classes. Le classifieur [27] est basé sur une analyse en composantes principales à noyaux combiné avec une analyse discriminante quadratique (ou régularisé) et donc considère à la fois des informations inter et intra-classes. Les temps de calculs des matrices de Gram associées aux trois noyaux sont donnés dans Tab. 1(bas). Tab. 1(haut) montre les matrices de confusions obtenues avec chaque approche. Le noyau $K_{combine}$ présente de très bons résultats avec un peu de confusion sur les tasses. Notons à ce sujet qu'il existe des tasses trouées et non trouées (Fig. 4a). L'existence de trous est donc faiblement discriminant sur ce type d'objet. Le noyau $K_{pondere}$ présente également de bons résultats, mais avec une forte confusion sur les outils. Cette confusion provient de la construction des sacs avec peu de chemins qui n'est pas suffisante pour une bonne séparation sur ce type d'objet. Les meilleures performances de $K_{combine}$ sur cette classe sont dues au noyau K_{chemin_ferme} qui différencie fortement les outils (sans trous) des autres classes. Le noyau sur marches aléatoires montre de bons résultats avec des confusions au niveau des outils et des tasses. La confusion sur les tasses est due aux arbres de recouvrement qui conduisent à une perte d'information sur les faces des graphes.

Classes	(1)	(2)	(3)	(4)	(5)
$K_{combine}$					
(1)	8	2	1		
(2)		11			
(3)			11		
(4)				11	
(5)	1				10
$K_{pondere}$					
(1)	7	4			
(2)		11			
(3)		2	9		
(4)				11	
(5)	1	3	2		5
Marches aléatoires					
(1)	4	6			1
(2)	2	8	1		
(3)			10	1	
(4)				11	
(5)		2		1	8

Noyaux	Temps
$K_{combine}$	19s
$K_{pondere}$	8s
Marches aléatoires	9min

TAB. 1: En haut, les matrices de confusions des 5 classes de formes : (1) Tasses, (2) Clés, (3) Ciseaux, (4) Silhouettes et (5) Outils. En bas, les temps de calculs des matrices de Gram pour chaque noyau.

7 Conclusion

Nous avons défini dans ce papier un noyau défini positif pour la classification de formes qui est robuste au bruit et prend les trous en compte. Les expériences montrent que cette prise en compte de caractéristiques topologiques comme les trous améliore les performances de classification. Dans le futur, nous pensons améliorer la sélection des chemins et la combinaison des résultats au sein des noyaux sur sacs.

Références

- [1] Xiang Bai and Jan Latecki. Path Similarity Skeleton Graph Matching. *IEEE PAMI*, 30(7), 2008.
- [2] Christian Berg, Jens Peter Reus Christensen, and Paul Ressel. *Harmonic Analysis on Semigroups*. Springer-Verlag, 1984.
- [3] Horst Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(8) :689–694, 1997.
- [4] I. DiMatteo, C.R. Genovese, and R.E. Kass. Bayesian curve fitting with free-knot splines. *Biometrika*, 88 :1055–1071, 2001.
- [5] F. X. Dupé and L. Brun. Edition within a graph kernel framework for shape recognition. In *GbRPR 2009*, pages 11–20, 2009.
- [6] F. X. Dupé and L. Brun. Tree covering within a graph kernel framework for shape classification. In *ICIAP 2009*, 2009. accepted.
- [7] Wooi-Boon Goh. Strategies for shape matching using skeletons. *Computer Vision and Image Understanding*, 110 :326–345, 2008.
- [8] Leo Grady. Random Walks for Image Segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11) :1768–1783, 2006.
- [9] B. Haasdonk and E. Pekalska. Indefinite kernel fisher discriminant. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4, Dec. 2008.
- [10] Bernard Haasdonk. Feature space interpretation of svms with indefinite kernels. *IEEE Trans. on PAMI*, 27(4) :482–492, April 2005.
- [11] David Haussler. Convolution kernels on discrete structures. Technical report, Department of Computer Science, University of California at Santa Cruz, 1999.
- [12] Heiko Hoffmann. Kernel PCA for novelty detection. *Pattern Recognition*, 40(3) :863–874, 2007.
- [13] Tamás Horváth. Cyclic pattern kernels revisited. In *PAKDD*, pages 791–801. Springer, 2005.
- [14] Hisashi Kashima, Koji Tsuda, and Akihiro Inokuchi. Marginalized kernel between labeled graphs. In *In Proc. of the Twentieth International conference on machine Learning*, 2003.
- [15] LEMS. shapes databases. <http://www.lems.brown.edu/vision/software/>.
- [16] Fernand Meyer. Topographic distance and watershed lines. *Signal Proc.*, 38(1), 1994.
- [17] M. Neuhaus and H. Bunke. *Bridging the Gap between Graph Edit Distance and Kernel Machines*, volume 68 of *Machine Perception and Artificial Intelligence*. World Scientific, September 2007.
- [18] M. Pelillo, K. Siddiqi, and S.W. Zucker. Matching hierarchical structures using association graphs. *IEEE Trans. on PAMI*, 21(11) :1105–1120, Nov 1999.
- [19] C. Di Ruberto. Recognition of shapes by attributed skeletal graphs. *Pattern Recognition*, 37(1) :21–31, 2004.
- [20] T.B. Sebastian, P.N. Klein, and B.B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Trans. on PAMI*, 26(5) :550–571, 2004.
- [21] Kaleem Siddiqi, Sylvain Bouix, Allen Tannenbaum, and Steven W. Zucker. Hamilton-jacobi skeletons. *International Journal of Computer Vision*, 48(3) :215–231, 2002.
- [22] Kaleem Siddiqi, Ali Shokoufandeh, Sven J. Dickinson, and Steven W. Zucker. Shock graphs and shape matching. *Int. J. Comput. Vision*, 35(1) :13–32, 1999.

- [23] Frédéric Suard, Alain Rakotomamonjy, and Abdelaziz Bensrhair. Kernel on bag of paths for measuring similarity of shapes. In *European Symposium on Artificial Neural Networks*, Bruges-Belgique, April 2007.
- [24] Frédéric Suard, Alain Rakotomamonjy, and Abdelaziz Bensrhair. Mining shock graphs with kernels. Technical report, LITIS, 2006. <http://hal.archives-ouvertes.fr/hal-00121988/en/>.
- [25] Andrea Torsello and Edwin R. Hancock. A skeletal measure of 2d shape similarity. *CVIU*, 95 :1–29, 2004.
- [26] S.V.N. Vishwanathan, Karsten M. Borgwardt, Imre Risi Kondor, and Nicol N. Schraudolph. Graph kernels. *Journal of Machine Learning Research*, 9 :1–37, 2008.
- [27] Jie Wang, K.N. Plataniotis, Juwei Lu, and A.N. Venetsanopoulos. Kernel quadratic discriminant for small sample size problem. *Pattern Recognition*, 41(5) :1528–1538, 2008.