

Approches structurales en analyse d'images et reconnaissance des formes




Jean-Yves Ramel
Laboratoire Informatique de Tours - FRANCE

Plan

- Segmentation & Représentation structurale d'images
 - Segmentation, localisation d'éléments de contenu
 - Extraction de primitives et signatures structurales
 - Représentation des relations spatiales et du voisinage
- Méthodes structurales d'analyse et reconnaissance de contenus
 - Méthodes statistiques vs structurales
 - Comparaison de chaînes
 - Comparaison de graphes
- Conclusion


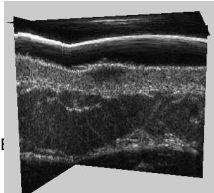

Partie 1 :

SEGMENTATION & EXTRACTION DE PRIMITIVES STRUCTURELLES

M2R Informatique Tours - Indexation de documents multimédias 

Qu'est ce qu'une segmentation ?

- Extraction, localisation d'objet (EdC ou de primitives)
- Notion de régions, primitives ou blocs :
 - Objet 1, objet 2 ...
 - fond - formes (binarisation, contour, région, ...)
 - Formes 1 - Formes 2 (écriture, partitions musicales)
- Notion de couches superposées :
 - couche texte, formes fines, formes pleines
 - niveau lexical, syntaxique, sémantique
 - Arrière plan, premier plan, ...
- Segmentation = séparation des EdC selon un (des) critère(s)

4

Segmenter une image : déterminer sa structure

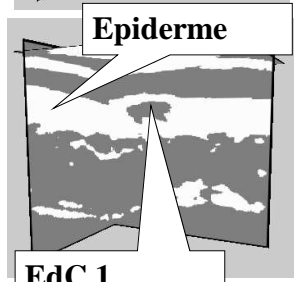
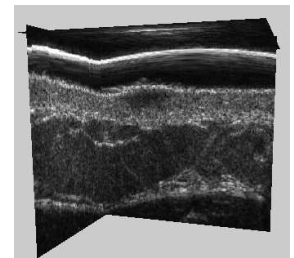
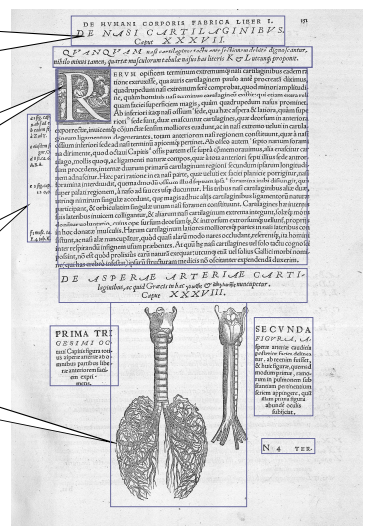
Titre, haut de page

Lettrine

Note en marge

Illustration

EdC



Epiderme

EdC 1

Impact de la segmentation ?

- Une bonne segmentation est déjà une étape importante pour reconnaissance !!!!
 - « c'est toujours la segmentation qui pose problème »
- Une segmentation est un traitement **irréversible** car c'est le résultat d'une **interprétation** suivant un **critère** et une **méthode**.
- Les images non segmentées conservent toute l'information. Toute segmentation est le produit d'un choix d'une méthode et de ses paramètres !
- Plusieurs méthodes de segmentation :
 - à partir de l'analyse de formes
 - Approche contours
 - Approche régions
 - à partir de connaissances → cf Ontologies !!!

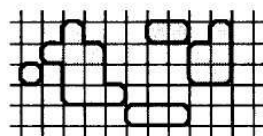
Approche structurelle de segmentation Structures des images, objets, EdC

- **Structure physique : disposition des contenus**
 - Décomposition d'une image : agencement des objets, des régions, des EdC, des espacements, ...
 - Taille, forme, couleur des EdC : types, polices de caractères, ...
- **Structure logique (ou atlas) : enchaînement et organisation**
 - Liée à la sémantique, au contenu de l'image
 - Liée au message véhiculé par l'image
 - Hiérarchie d'objets, enchaînements des primitives
- **Ces deux structures sont fortement liées**

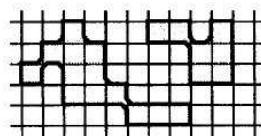
7

Représentation structurelle : Approche Régions

- Analyse **des composantes connexes**
 - Un ensemble S de pixels est une composante 4-connexe si et seulement si pour toute paire de pixels $P, Q \in S$, il existe un 4-chemin p_1, p_2, \dots, p_n tel que $p_1=P$ et $p_n=Q$ et tous les $p_i \in S$.
 - Un ensemble S de pixels est une composante 8-connexe si et seulement si pour toute paire de pixels $P, Q \in S$, il existe un 8-chemin p_1, p_2, \dots, p_n tel que $p_1=P$ et $p_n=Q$ et tous les $p_i \in S$.



5 composantes 4-connexes

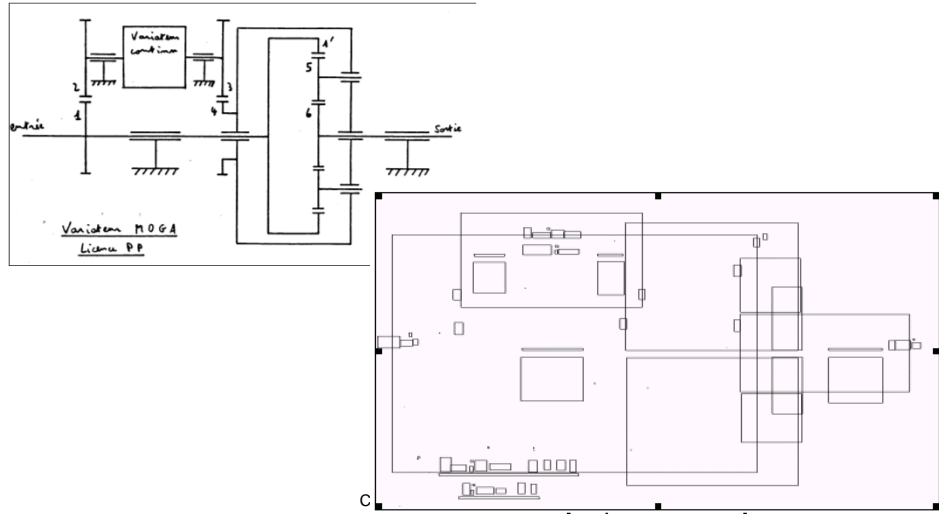


2 composantes 8-connexes

8

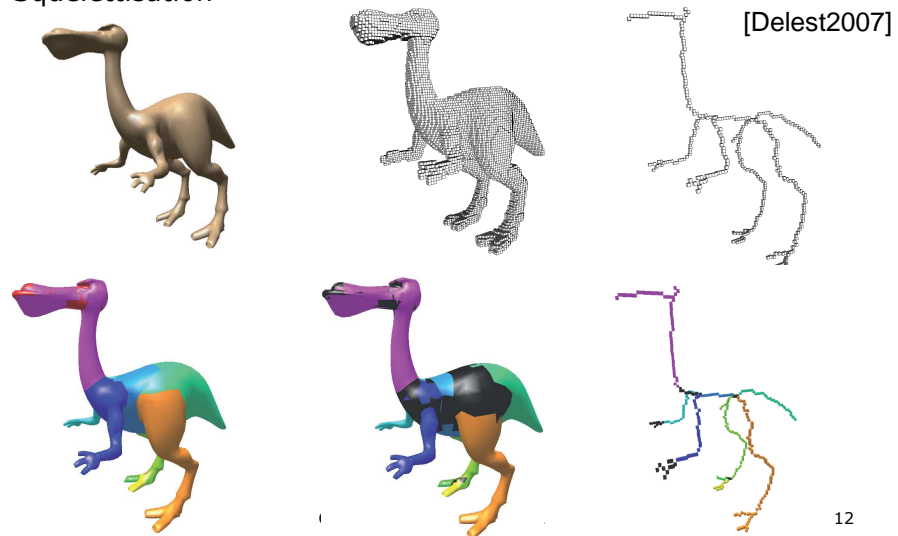
Application : Séparation Texte - Graphique

- Découpage en couches superposées



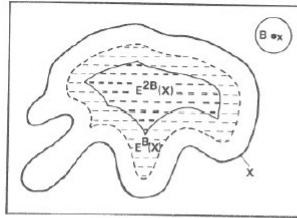
Description structurelles : Squelette

- Squelettisation



Squelettisation

Traitements itératifs (exemple : érosions itératives)

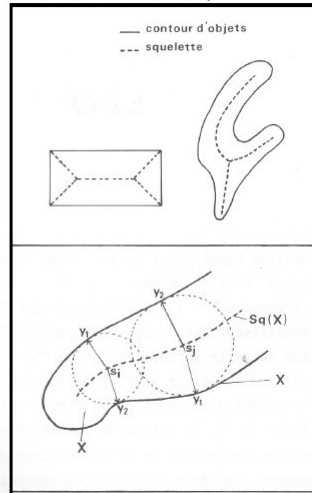


Squelettes

L'ensemble des points situés à équidistance de deux points du bord

L'ensemble des centres des boules maximales incluses dans la forme

Les squelettes sont calculées par érosion morphologique



13

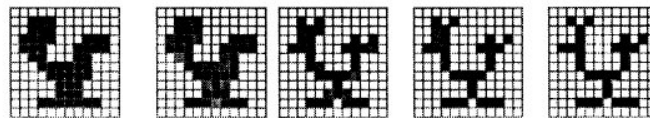


Illustration de la construction du squelette: composante initiale X, amincissement par les masques M_1 à M_4 , résultat des deux premières itérations et squelette obtenu $Sq(X)$

Élagage

La squelettisation produit souvent des formes bruitées caractérisées par des barbules se greffant sur le squelette proprement dit. L'élagage est une opération morphologique qui permet de supprimer les barbules.

Le procédé consiste à éliminer de manière itérative des pixels terminaux au moyen des masques.

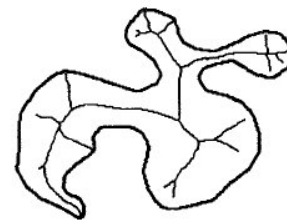
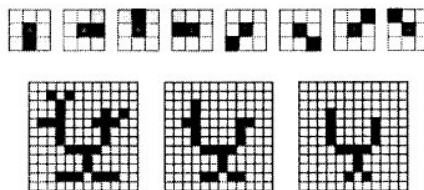
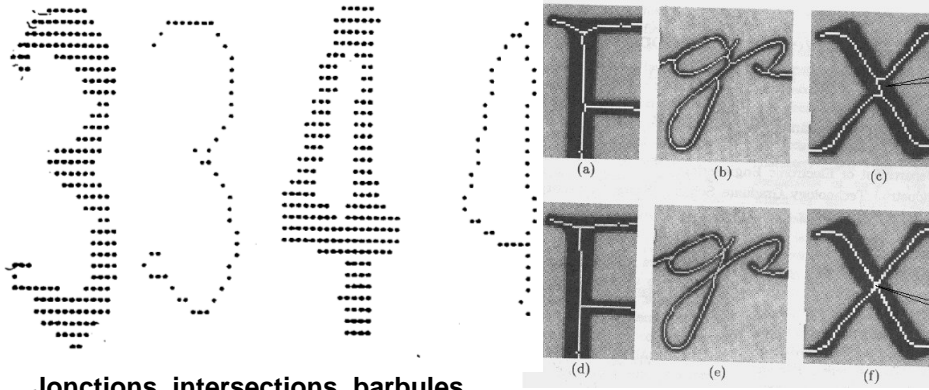


Illustration de l'élagage d'un squelette obtenu en deux itérations successives

Problème du squelette . . .



Jonctions, intersections, barbules, ...

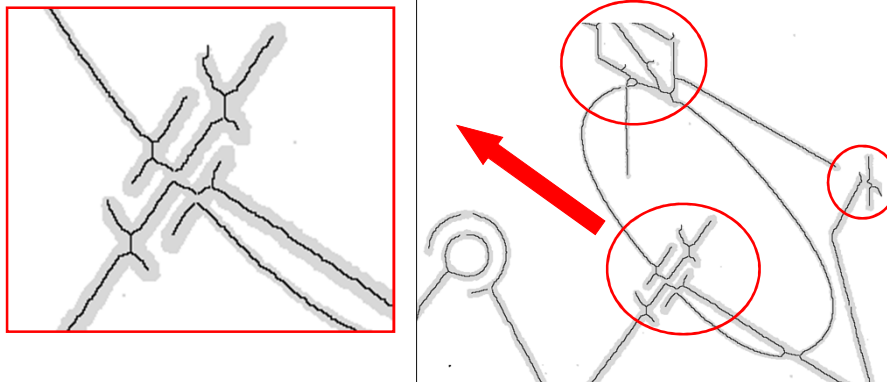
Lenteurs, complexité, paramétrages, ...

Mais très souvent utilisé (méthode structurelle)

15

Problème du squelette . . .

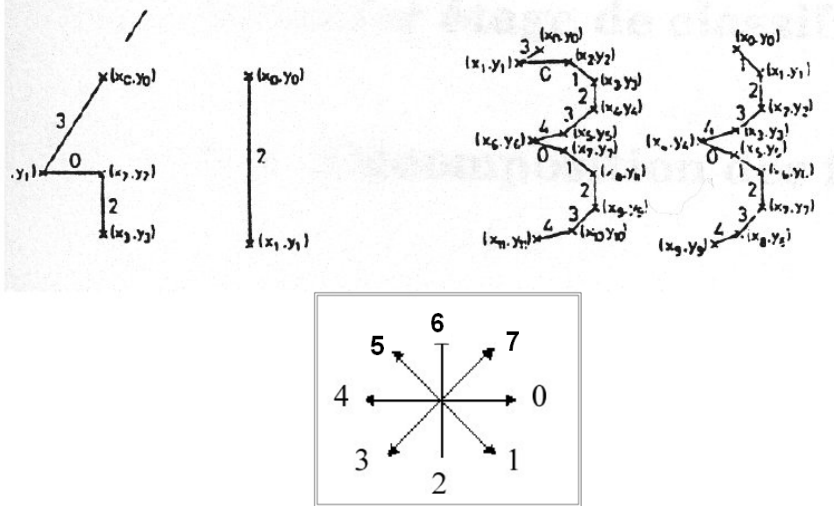
Barbules, jonctions et croisements



16

Squelette, et après ?

Chainage (suivi) du squelette → directions de Freeman



Représentation structurales des contours

Extraction des contours ...



Et après ???



Vers les EdC par l'étude des contours

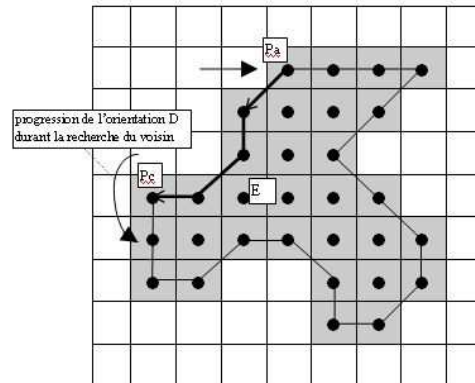
- Suivi de contour → chaîne de Freeman →

3	2	1
4	Pc	0
5	6	7

Image naturelle (NdG) !!!



Image binaire



Etude des contours : Freeman et suivi

$$D_d = D_1 - D_2$$

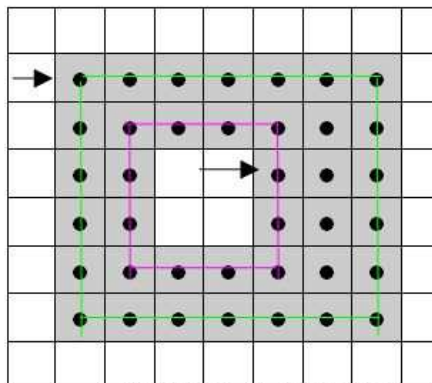
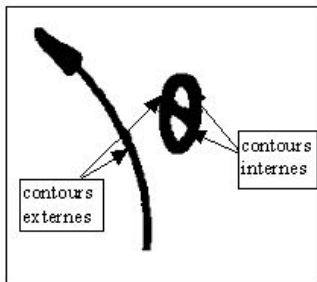
Si $(D_1 - D_2) > 4$ alors $D_d = D_d - 8$

Si $(D_1 - D_2) < -3$ alors $D_d = D_d + 8$

Le retour-arrière ne se produisant jamais, $(D_1 - D_2) = -4$ ne se réalise jamais.

Avantages et inconvénients

- Point de départ
- Changement d'échelle

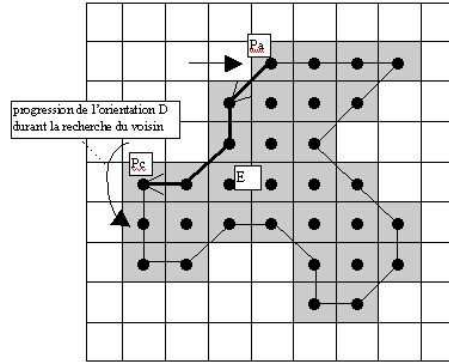
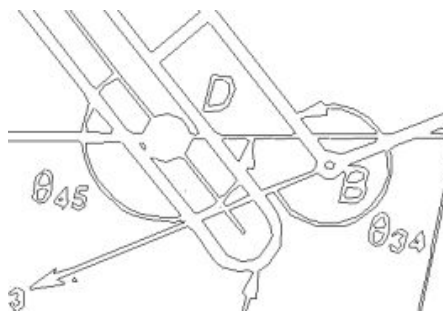


Contour interne : 664442220006 => 0-200-200-200-20 => $S_d = -8$

Contour externe : 6666600000022222444444 => 0000200002000020000002 => $S_d = 8$

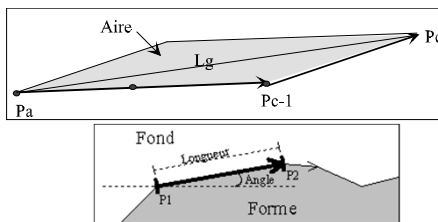
Approximation polygonale → Vectorisation

- Des contours/squelettes aux vecteurs

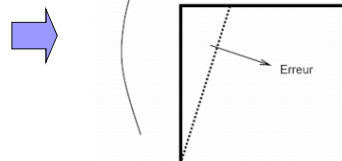


Approximation polygonale → Vectorisation

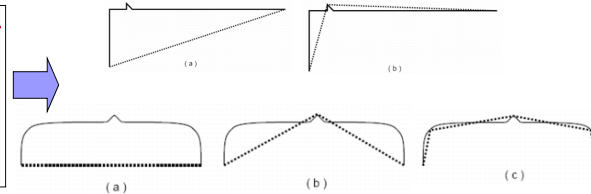
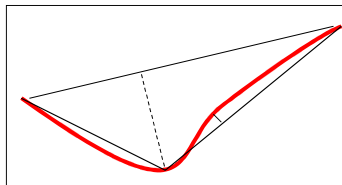
- Méthodes itératives [Wall 84]



Sens de parcours de la chaîne



- Méthodes Récursives

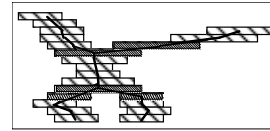


Approximation polygonale → Vectorisation

- Et bien d'autres méthodes...

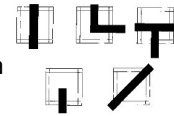
- RLE+LAG

- Run Length Encoding - Line adjacency Graph
- Ramachandra80, Pavlidis86



- Line tracker

- On se cale sur les contours d'une ligne et on la suit en déplaçant un segment qui reste perpendiculaire à la direction de la ligne
- Fukada



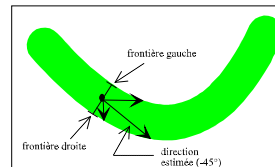
- Suivi de traits

- **Orthogonal Zig-Zag** : Prolongement de longueur maximale suivant les directions de Freeman - Dori
- Système ANON - Joseph



- Maillage

- Lin, Shimotsuji, Minoh, Sakai, Vaxivière



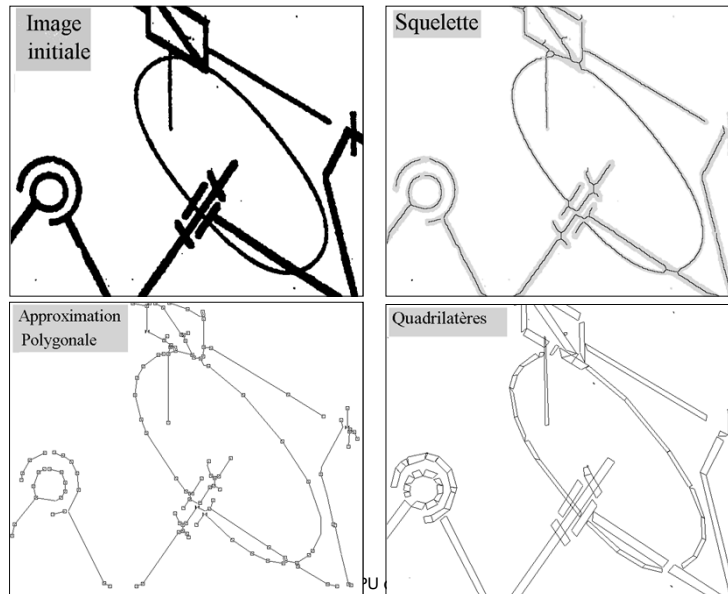
Approximation polygonale → Vectorisation

- Appariement de contours :

```

Construction_Vecteurs();
Répéter
    nb_fusion = Fusion_Vecteurs();
Jusqu'à nb_fusion = 0;
Construction_Quadrilateres();
Répéter
    Ordonnancement_Quadrilateres();
    nb_fusion = Fusion_Quadrilateres();
Jusqu'à nb_fusion = 0;
  
```

Résultats et DEMO



25

Relations entre primitives, objets, EdC

- Comment décrire les relations entre EdC ou primitives ?

- Grammaires

- Pour les séquences 1D de primitives

- Les alphabets :



Primitives [Coueignoux].

voici HOMERE

Texte source.

voici HOMERE

Décomposition du texte source

- Pas complètement adaptées aux images (2D ou 3D)

Graphes : Un pouvoir de représentation élevé...

- Topologie :
 - Nœuds = éléments ou parties
 - Arcs = relations
- Attributs :
 - Statistiques : observations de points, distributions, ...
 - Géométriques : métriques (distances, angles, courbures)
 - Positions : absolues ou relatives
 - Indices visuels : signes distinctifs
- Assurant :
 - Stabilité (invariance)
 - Tolérance : bruits, variations
 - Séparation des classes
- Mais
 - Symboliques VS numériques
 - Discrétisation

27

Représentation d'images à base de graphes

Modélisation d'objets à l'aide de graphes

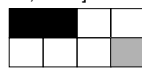
Multitudes d'applications : Molécules, objets de conception, documents, ...

Graphe de pixels

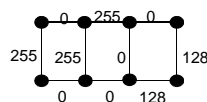
Les pixels → les sommets du graphe

Les arcs → les valeurs absolue de la différence de niveaux de gris

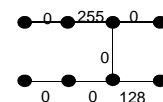
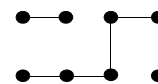
[Morris, 1986]



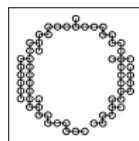
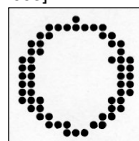
Image



Graphe attribué

Arbre couvrant
de poids minimalSuppression des arêtes
les plus coûteuses

[Franco, 2003]



Problème

Les graphes de pixels sont bien souvent de tailles trop importantes pour être analysés

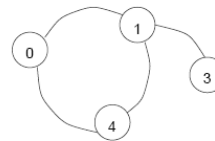
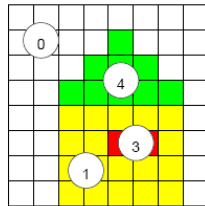
28

Représentation d'images à base de graphes

Graphe d'adjacence de régions

Graphe non orienté où les noeuds correspondent aux régions, \exists arête entre 2 noeuds ssi les 2 régions correspondantes sont adjacentes.

A utiliser avec une description des régions.



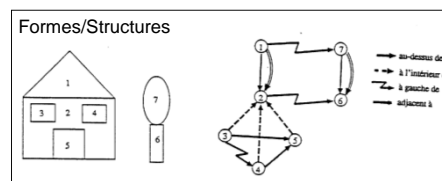
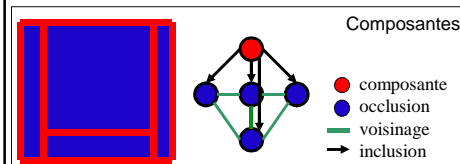
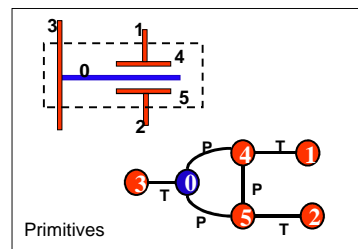
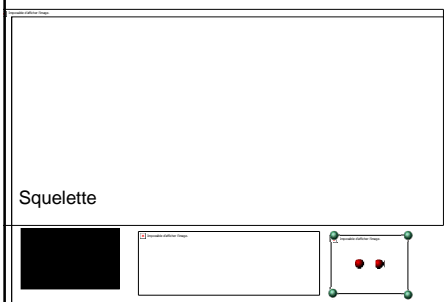
[Brun, 2007]

+ fusion de régions adjacentes (contraction d'une arête + suppression éventuelle d'arêtes multiples)

- découpe (division d'une région en 2 ?)

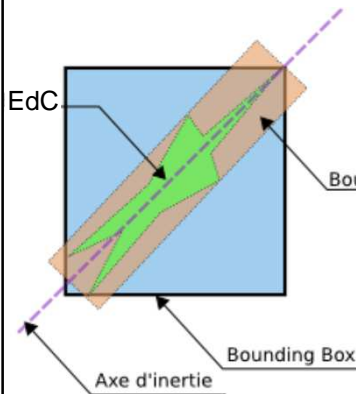
Représentation d'images à base de graphes

Graphe de Primitives / Formes / Structures



Un exemple de représentation

Image → Région/primitives → Graphe d'EdC



Arc = Relation entre EdC

- Distance minimale entre les deux EdC
- Angle inter EdC
- Relation : Gauche, Droite, Dessus, Dessous, Inclus, Partiellement inclus, L, T, P, X, S, Indefini

Noeud = EdC

- Type d'EdC
- Centre $c : (X, Y)$ de la Bounding Box
- Bounding Box
- Bounding Rectangle BR : $(P1, P2, P3, P4)$
- Angle principal = angle du rectangle englobant
- Densité des niveaux de gris de la BB
- Couleur : couleur moyenne de l'EdC
- Liste des sous EdC primaires que contient l'EdC
- Nombre de EdC primaires
- Taux de confiance de la reconnaissance

31

Représentation d'images à base de graphes

Une représentation générique → Graphe d'EdC

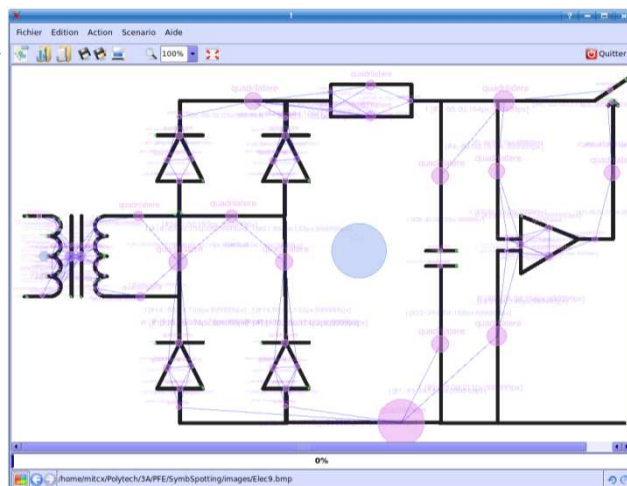
Graphe d'EdC →

Primaires

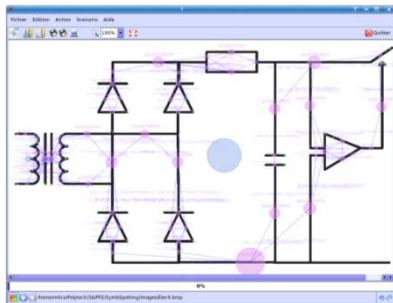
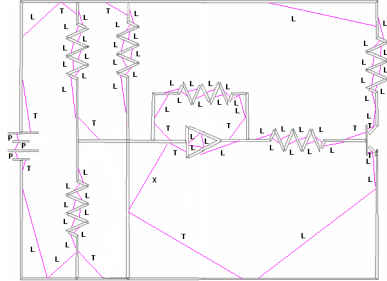
- Composantes connexes
- Vecteurs
- Quadrilatères

Qui évoluent

- Caractères
- Mots
- Lettrines
- Triangles
- Diodes
- ...



Un exemple de représentation



```

<Graph id="Symbole4">
  <node id="node0">
    <attr name="forme">string-Quad</string></attr>
    <attr name="x1">string-232</string></attr>
    <attr name="y1">string-497</string></attr>
    <attr name="x1f">string-231</string></attr>
    <attr name="y1f">string-417</string></attr>
    <attr name="x2">string-227</string></attr>
    <attr name="y2">string-418</string></attr>
    <attr name="x2f">string-229</string></attr>
    <attr name="y2f">string-498</string></attr>
    <attr name="angle1">string-90</string></attr>
    <attr name="angle2">string-91</string></attr>
    <attr name="thickness1">string-8</string></attr>
    <attr name="thickness2">string-7</string></attr>
    <attr name="length">string-81</string></attr>
    <attr name="score">string-0.703</string></attr>
  </node>
  ...
  ...
  <edge id="edge1" from="node0" to="node1">
    <attr name="angle">string-89</string></attr>
    <attr name="type">string-T</string></attr>
    <attr name="score">string-0.409</string></attr>
  </edge>
  <edge id="edge2" from="node1" to="node2">
    <attr name="angle">string-46</string></attr>
    <attr name="type">string-L</string></attr>
    <attr name="score">string-0.388</string></attr>
  </edge>
  ...
  ...

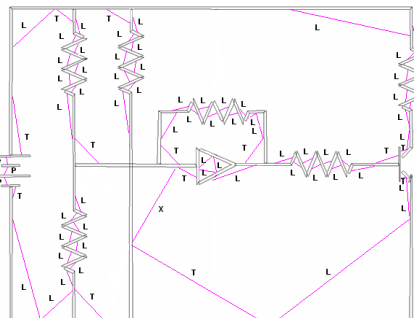
```

Segmentation d'images à base de graphes

Spotting de régions d'intérêts dans les graphes

Une première ébauche

- Attribuer des scores aux nœuds et arcs du graphes
- A l'aide d'heuristiques
- Via un module d'apprentissage



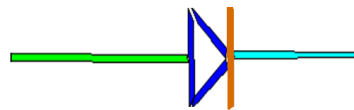
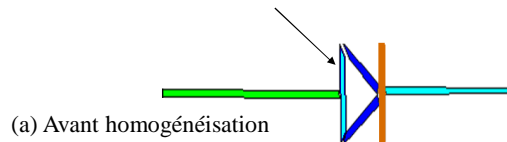
Heuristiques employées

- H1 – Les symboles se composent de petits segments comparés à d'autres éléments des schémas
- H2 – Les segments constituant un symbole ont des longueurs similaires
- H3 – Les symboles comportent souvent des boucles fermées
- H4 – Des symboles se composent souvent de segments parallèles
- H5 – Un segment de symbole est rarement relié à plus de 3 autres segments
- H6 – Deux segments voisins avec un angle relatif loin de 90° correspondent souvent à une partie de symbole

Segmentation d'images à base de graphes

Spotting de régions d'intérêts dans les graphes

Propagation des scores maximum à tous les nœuds dans les boucles

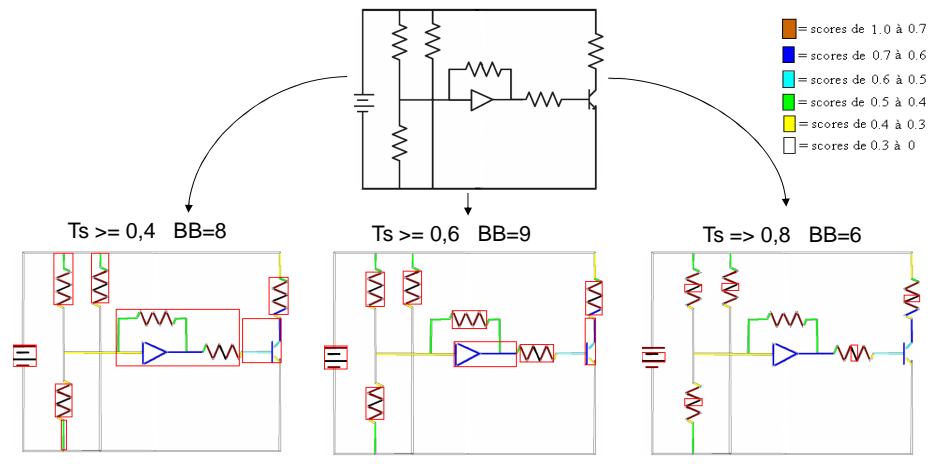


(b) Après homogénéisation le score de 0.7 a été propagé à tous les quadrilatères appartenant à la boucle

Segmentation d'images à base de graphes

Spotting de régions d'intérêts dans les graphes

Extraction des zones d'intérêt et sous-graphes associés



Segmentation d'images à base de graphes

Spotting de régions d'intérêts dans les graphes

Expérimentations sur différents types de documents

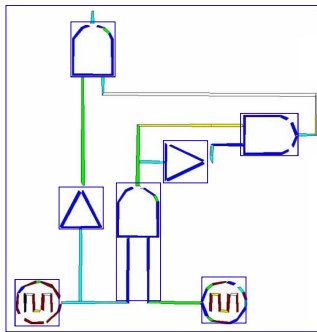
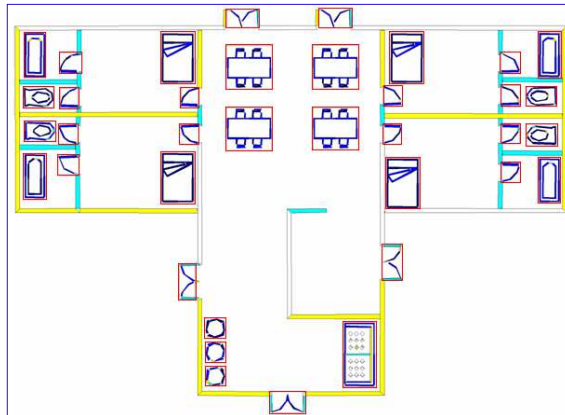


Diagramme logique $T_s \geq 0,6$



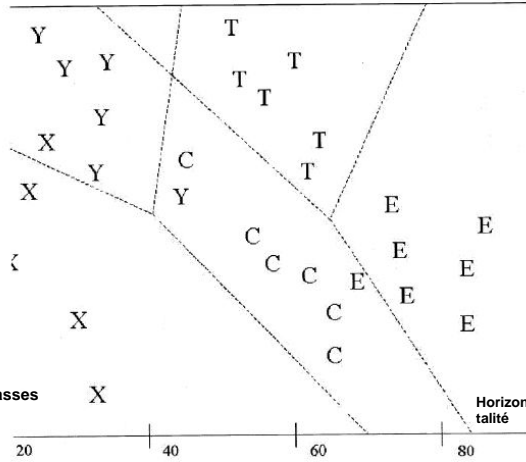
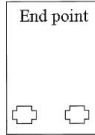
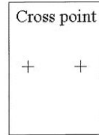
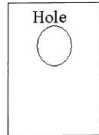
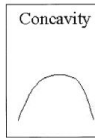
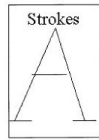
Plan d'architecture $T_s \geq 0,5$

37

Partie 2 :

Méthodes structurales de Reconnaissance d'éléments de contenu

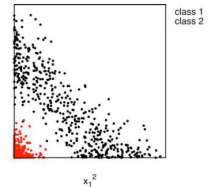
Méthodes statistiques : exemple



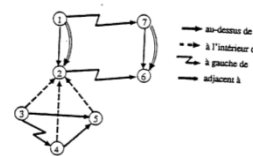
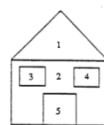
- **Apprentissage supervisé ou non :**
 - Sélection des paramètres
 - Création des classes et frontières entre classes
- **Théorie de la décision de Bayes :**
 - distributions, classes et paramètres connus
- **Techniques non paramétriques :**
 - KPPV , e -voisins...

Méthodes statistiques vs structurelles ?

- **Méthodes statistiques**
 - Création des classes et frontières entre classes
 - Il existe des outils statistiques qui analysent la séparabilité des classes et permet de conclure sur la pertinence des descripteurs choisis
 - Multitude de classifieurs
 - Apprentissage (base, supervisé ou non)



- **Méthodes structurelles**
 - Prise en compte du contexte
 - Reconnaissance partielle, locale et incrémentale
 - Choix des modèles
 - Choix des caractéristiques
 - Temps de calcul, complexité ?
 - Apprentissage ?



Statistiques vs structurelles ?

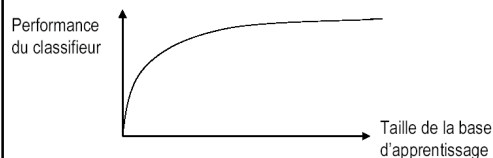
- Quelques évidences bonnes à rappeler
 - Évidence 1 : sans de bonnes caractéristiques, aucun classifieur ne donnera de bonnes performances
 - Évidence 2 : Avec des caractéristiques raisonnablement pertinentes, tous les classifieurs ne donnent pas les mêmes résultats
 - Évidence 3 : Les données observées (base d'apprentissage) conditionnent complètement les performances
 - Évidence 4 : En cas d'échec, il faut remettre en question les descripteurs, les données observées et enfin le type de classifieur (ou son implémentation)

41

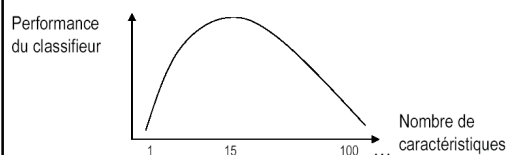
Statistiques vs structurelles ?

- Quelques évidences bonnes à rappeler
 - Choix et nombre de caractéristiques
 - Complexité

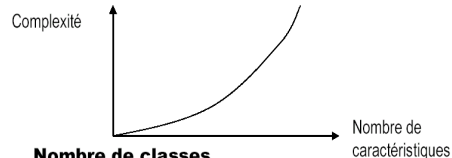
Statistique suffisante



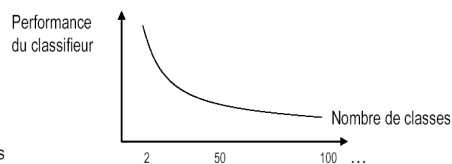
Malédiction de la dimensionalité



Complexité vs nombre de caractéristiques

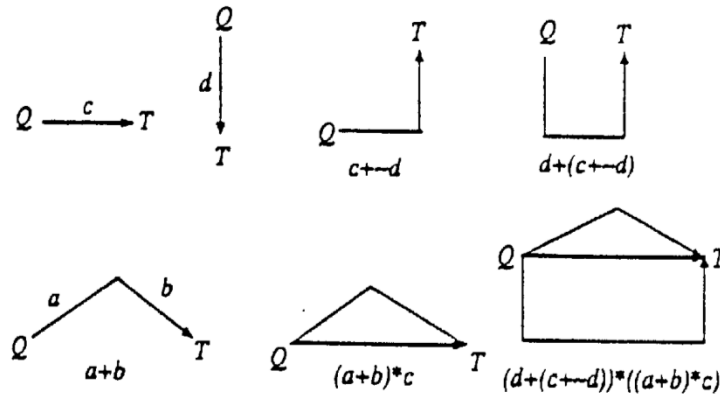


Nombre de classes



Reconnaissance syntaxique

■ Méthodes syntaxiques à base de grammaires



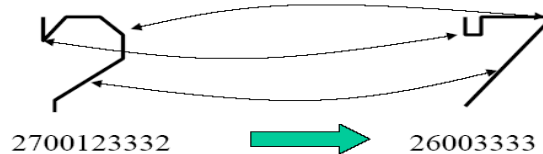
Cours de l'EPU de Tours - DEA

43

Comparaison de chaînes

■ Distance d'édition : calculer le nombre de transformations pour transformer une chaîne en une autre

- Transformations possibles :
 - insertion d'un élément
 - suppression d'un élément
 - échanger un élément par un autre



Distance = coût(7→6) + coût(suppr1) + coût(2→3) + coût(suppr2)

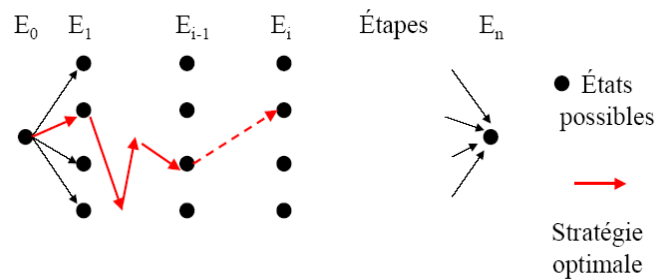
Cours de l'EPU de Tours - DEA

44

Comparaison de chaînes

- **Théorème d'optimalité de Bellman**: une stratégie optimale est nécessairement constituée de sous-stratégies optimales
- Application aux graphes, aux chaînes, dans la recherche d'un chemin dans un réseau...

$$Opt(E_0, E_i) = Opt(E_0, E_{i-1}) + Opt(E_{i-1}, E_i)$$



45

Comparaison de chaînes

Coût minimal $C[i,j]$ de transformation du 1^{er} au j^{ème} élément d'une chaîne x vers une chaîne y du 1^{er} au j^{ème} élément

=

Coût minimal jusqu'à l'élément précédent de la chaîne x ou de y ou des deux

$$\min\{C[i-1,j], C[i,j-1], C[i-1,j-1]\}$$

+

Le coût $\delta(x[i],y[j])$ de transformation de $x[i]$ en $y[j]$

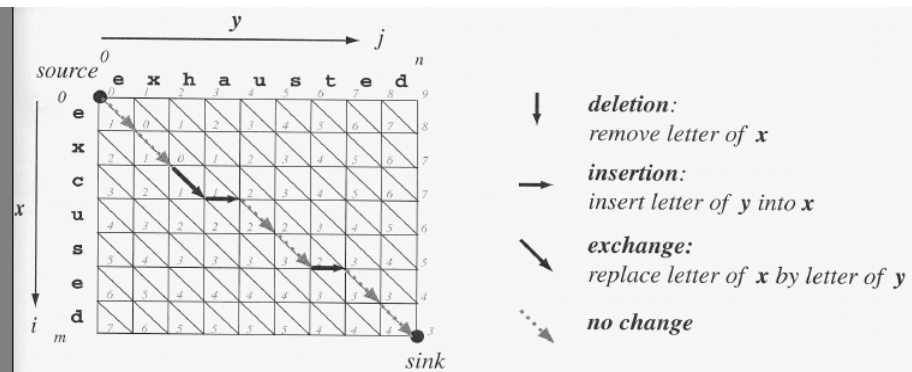
$$C[i,j] = \min\{C[i-1,j], C[i,j-1], C[i-1,j-1]\} + \delta(x[i],y[j])$$

Comparaison de chaînes

```

1 begin initialize  $A, x, y, m \leftarrow \text{length}[x], n \leftarrow \text{length}[y]$ 
2    $C[0, 0] \leftarrow 0$ 
3    $i \leftarrow 0$ 
4   do  $i \leftarrow i + 1$ 
5      $C[i, 0] \leftarrow i$ 
6   until  $i = m$ 
7    $j \leftarrow 0$ 
8   do  $j \leftarrow j + 1$ 
9      $C[0, j] \leftarrow j$ 
10  until  $j = n$ 
11   $i \leftarrow 0; j \leftarrow 0$ 
12  do  $i \leftarrow i + 1$ 
13    do  $j \leftarrow j + 1$ 
14       $C[i, j] = \min \left[ \underbrace{C[i - 1, j] + 1}_{\text{insertion}}, \underbrace{C[i, j - 1] + 1}_{\text{deletion}}, \underbrace{C[i - 1, j - 1] + 1 - \delta(x[i], y[j])}_{\text{no change/exchange}} \right]$ 
15    until  $j = n$ 
16  until  $i = m$ 
17  return  $C[m, n]$ 
18 end
    
```

Comparaison de chaînes



x	excused	source string	$C[0, 0] = 0$
	ex <u>h</u> used	substitute h for c	$C[3, 3] = 1$
	ex <u>h</u> aused	insert a	$C[3, 4] = 2$
	exhaust <u>e</u> d	insert t	$C[5, 7] = 3$
y	exhausted	target string	$C[7, 9] = 3$

Comparaison de chaînes

- **Proposition d'amélioration de la distance d'édition :**
- Autres transformations possibles :
 - insertion d'un élément
 - suppression d'un élément
 - échanger un élément par un autre
 - **Fusion d'éléments d'une chaîne**

Distance = coût(7→6) + coût(suppr1) + **coût(241→3)** + coût(suppr2)

- Distance d'édition cyclique (pour les contours) :
 - Duplication de la 2^e chaîne dans le tableau

49

Reconnaissance des Formes à base de graphes

Reconnaissance des formes → Mises en correspondance des graphes

Définition (appariement)

Un appariement de $G1 = (V1; E1)$ et $G2 = (V2; E2)$
 = une relation $m \subseteq V1 \times V2$ ($u1; u2 \in m$)
 ⇒ le sommet $u1$ est apparié au sommet $u2$

Différents types d'appariements [Solnon, 2007]

Appariement bijectif : cardinalité = (1; 1)

Appariement injectif : cardinalité = (1; 0..1)

Appariement univoque : cardinalité = (0..1; 0..1)

...

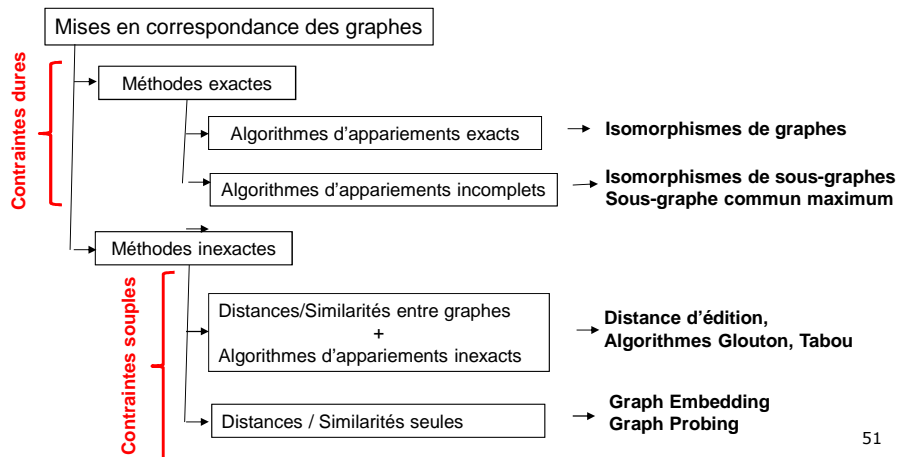
Appariement multivoque : cardinalité = (0..|V2|; 0..|V1|)

Complexité → Vers des méthodes sans appariement !

50

Reconnaissance des Formes à base de graphes

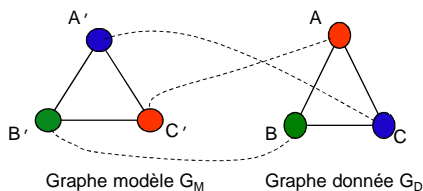
Reconnaissance des formes → Mises en correspondance des graphes
Le problème du mode de prise en compte des attributs



Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes dures

Problème de l'isomorphisme de graphes



Objectif

- Appariement bijectif
- Contraintes dures
- Possible sur de gros graphes

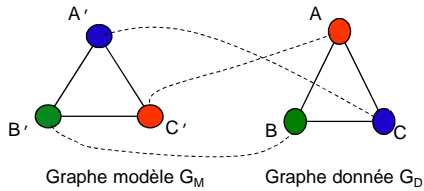
Problème

- Pas robustes aux bruits et distorsions

Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes dures

Problème de l'isomorphisme de graphes



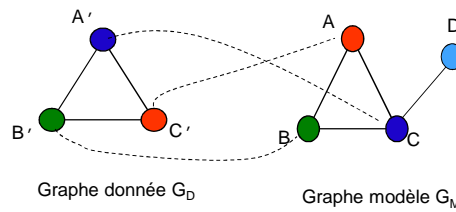
Objectif

- Appariement bijectif
- Contraintes dures
- Possible sur de gros graphes

Problème

- Pas robustes aux bruits et distorsions

Problème de l'isomorphisme de sous-graphe



Objectif

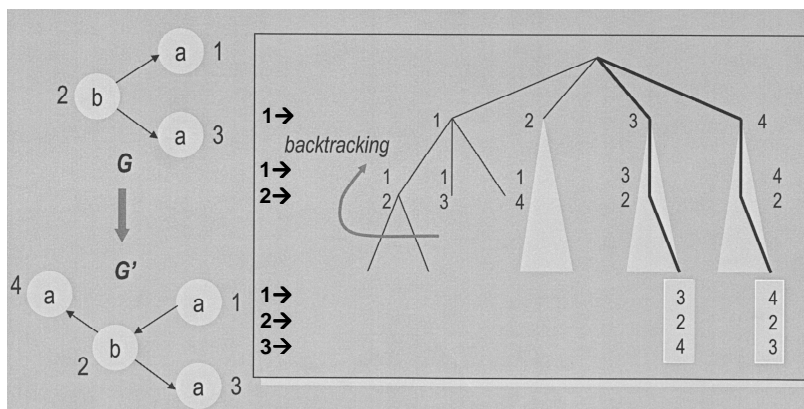
- Appariement injectif
- Contraintes dures
- NP-complet

Problème

- Résolution possible pour des graphes pas trop gros

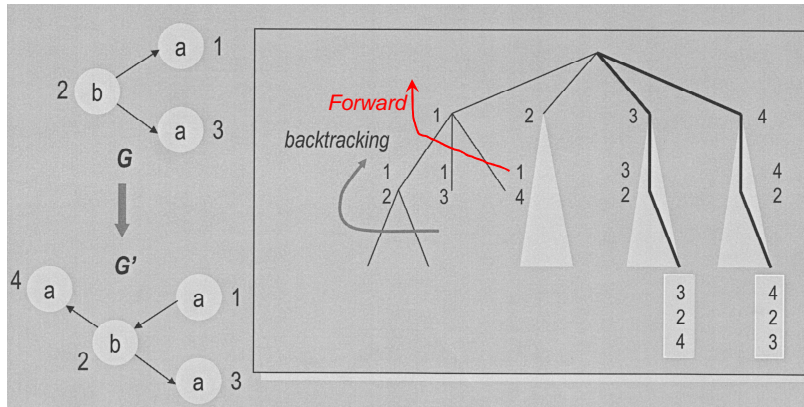
Reconnaissance des Formes à base de graphes

Parcours d'arbres (avec backtrack)



Reconnaissance des Formes à base de graphes

Parcours d'arbres (avec forward checking)



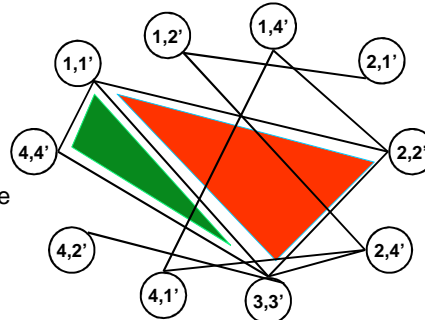
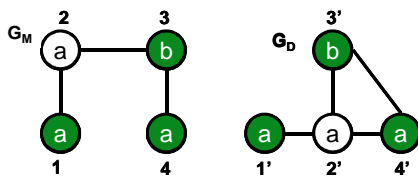
Ajout d'une fonction look-ahead() qui vérifie avant chaque association, l'existence d'une association possible au pas suivant.

Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes dures

Plus grand sous graphe-commun

Résolution à partir du graphe d'association (dans lequel les arcs représentent la compatibilité topologiques de l'appariement sommet à sommet)



Recherche de cliques maximisant un critère

Taille : **C1** [(1, 1'), (2, 2'), (3, 3')]

C2 [(1, 1'), (3, 3'), (4, 4')]

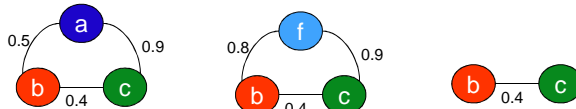
Problème :

- La complexité en temps ...
- Possible pour des graphes pas trop gros avec des attributs symboliques

Reconnaissance des Formes à base de graphes

Approches inexactes

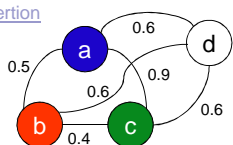
Plus grand sous-graphe (partiel) commun



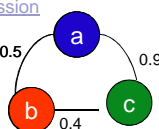
Distance d'édition de graphes

Trouver l'ensemble d'opérations le moins coûteux permettant de rendre les graphes isomorphes

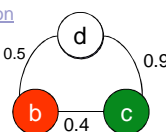
Insertion



suppression



substitution



Problème

Ces appariements sont univoques : chaque sommet ne peut être relié qu'à au plus un autre sommet alors que les objets peuvent être de granularités différentes

Reconnaissance des Formes à base de graphes

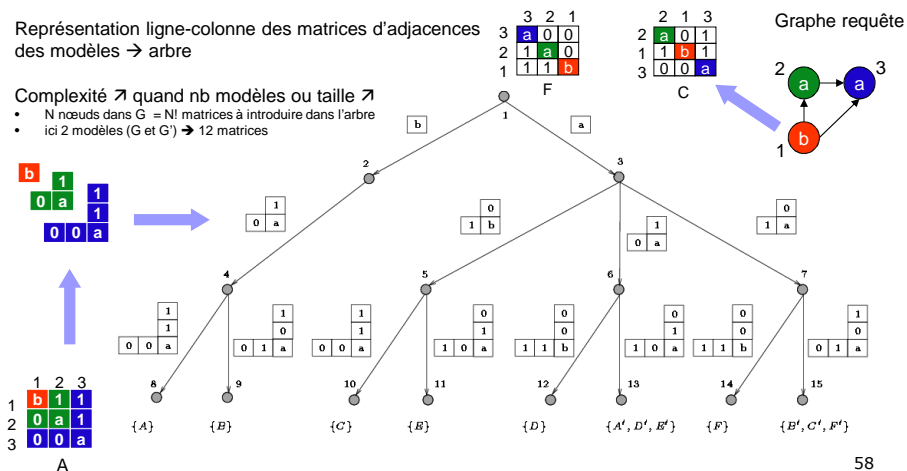
Appariements univoques – Contraintes dures

Recherche d'isomorphisme à partir des matrices d'adjacence [Bunke00]

Représentation ligne-colonne des matrices d'adjacences des modèles → arbre

Complexité \nearrow quand nb modèles ou taille \nearrow

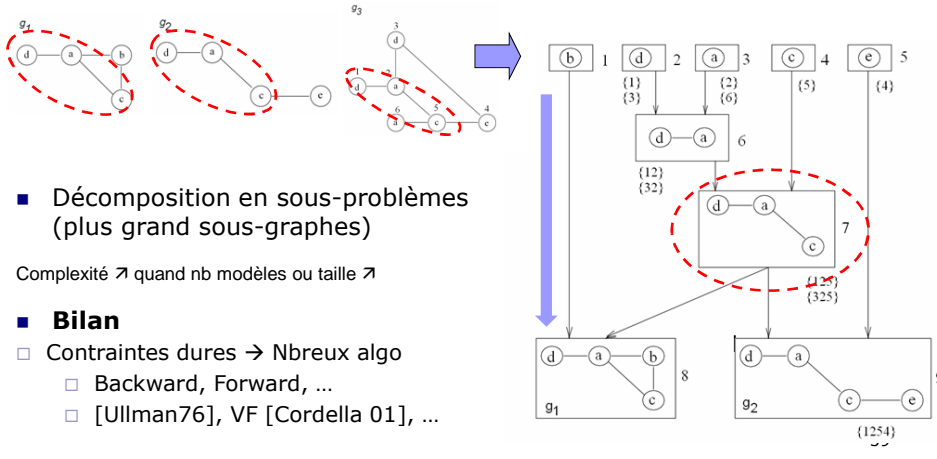
- N nœuds dans G = NI matrices à introduire dans l'arbre
- ici 2 modèles (G et G') → 12 matrices



Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes dures

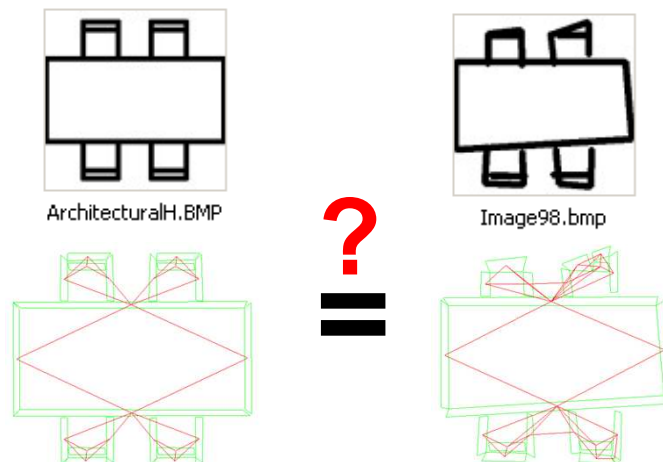
Plus grand sous graphe commun à partir d'arbre de décision [Messmer95]



Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes dures

TROP DURES...



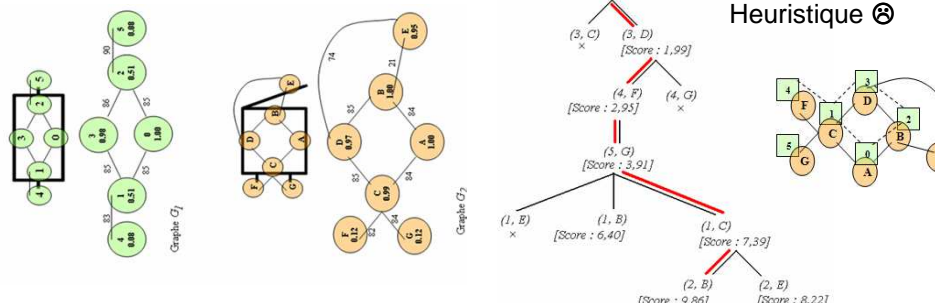
Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes SOUPLES

■ Contraintes souples

- Notion de similarité ≠ identique
- Exploration des possibilités...
Très couteux en temps...
- Matrice de similarité entre sommets

	A	B	C	D	E	F	G
0	1,00	1,00	0,99	0,97	0,95	0,12	0,12
1	0,51	0,51	0,51	0,52	0,54	0,61	0,61
2	0,51	0,51	0,51	0,52	0,54	0,56	0,61
3	0,98	0,98	0,98	0,99	0,97	0,14	0,14
4	0,08	0,08	0,08	0,11	0,13	0,96	0,96
5	0,08	0,08	0,08	0,11	0,13	0,96	0,96



Reconnaissance des Formes à base de graphes

Appariements univoques – Contraintes souples

Distance d'édition entre graphes [Bunke 99]

Coût : on associe pour chaque transformation (Insertion, suppression, substitution – arc ou nœud) un coût

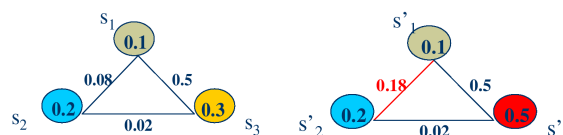
Séquence : l'ensemble d'opérations d'édition pour transformer un graphe en un autre

Erreur de correction : le coût globale d'une séquence transformation

Objectif : rechercher la séquence de transformation qui :

1. réalise un appariement (transformation) entre deux graphes
2. dont l'erreur de correction est minimal

Exemple

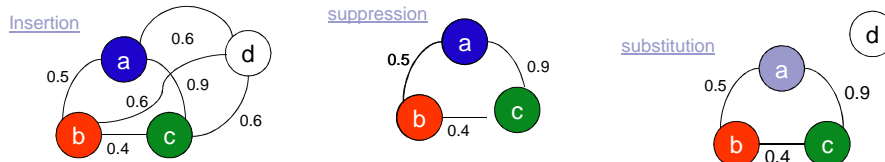


$$s_3 \rightarrow s'_3 \text{ [0.2]} \quad ; \quad (s_1, s_2) \rightarrow (s'_1, s'_2) \text{ [0.1]}$$

Cout(Séquence) = 0,3 : une substitution de sommet et une substitution d'arête. 62

Reconnaissance des Formes à base de graphes

Encore plus fort,
des appariements univoques aux appariements MULTIVOQUES



Ces appariements sont univoques → chaque sommet ne peut être relié qu'à au plus un autre sommet alors que les objets peuvent être de granularités différentes

Vers une nouvelle version de Distance d'Édition

- Toujours des contraintes souples
- Mais d'autres opérations deviennent possibles ?

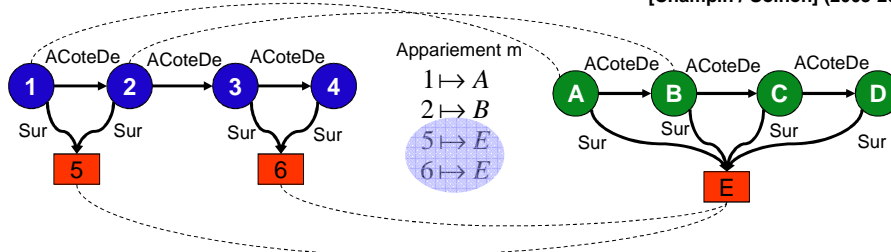
MERGE & SPLIT → Appariements multivoques ...

Reconnaissance des Formes à base de graphes

Appariements multivoques

Mesures de similarité de graphes

[Champin / Solnon] (2003-2005)



Problème : le choix / définition de mesures de similarité [Qureshi03]

$$Sc_{Mp} = \left[\underbrace{\sum_{i=1}^m (1 - \Delta V_i)}_{\text{Similarité nœud à nœud}} + \underbrace{\sum_{j=1}^n (1 - \Delta E_j)}_{\text{Similarité arc à arc}} - \underbrace{\left(\sum_{i=1}^k \omega_i + \sum_{j=1}^{\ell} \omega'_j \right)}_{\text{Pénalités pour les appariements multiples (splits)}] \right]$$

Reconnaissance des Formes à base de graphes

Appariements multivoques

Exploration d'appariements → un problème très combinatoire !

But = trouver $m \subseteq V_1 \times V_2$ qui maximise

$$\text{score}(m) = f(G_1 \cap_m G_2) - g(\text{splits}(m))$$

Problème NP-difficile → $2^{|V_1| \cdot |V_2|}$ combinaisons

Résolution par une approche complète ?

Structurer l'espace de recherche en treillis...

...mais la fonction score n'est pas monotone / inclusion

Limité à de très petits graphes

Résolution par des approches heuristiques

Glouton : construire rapidement un "assez bon" appariement

Tabou : améliorer un appariement en explorant son voisinage

65

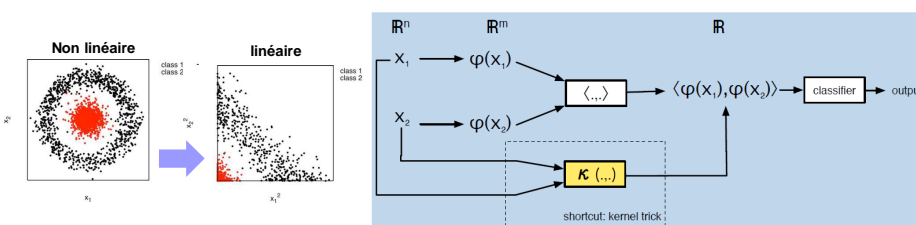
Reconnaissance des Formes à base de graphes

Pas d'appariement → Graph Kernel / Graph Embedding [Bunke09]

Combinaison intéressante des approches structurelles et statistiques

Graph Kernel

Fonction Kernel → $k : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ telle que $k(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$
avec x_1 et $x_2 \in \mathbb{R}^n$ et la fct $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}^m$ tq $m \geq n$



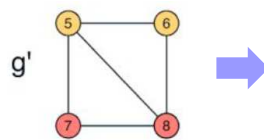
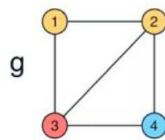
Graph Kernel → $k : G \times G \rightarrow \mathbb{R}$ tel que $k(g_1, g_2) = \langle \varphi(g_1), \varphi(g_2) \rangle$
avec $\varphi : G \rightarrow \mathbb{R}^n$

66

Reconnaissance des Formes à base de graphes

Graph Kernel / Graph Embedding [Bunke09]

Exemple



- $\varphi(g) = \frac{(2,1,1)}{4}$
- $\varphi(g') = \frac{(2,2,0)}{4}$
- $\kappa(g, g') = \frac{4+2}{16} = \frac{6}{16}$

Graph Embedding

$$\varphi : G \rightarrow R^n \quad \varphi(g) = (x_1, \dots, x_n)'$$

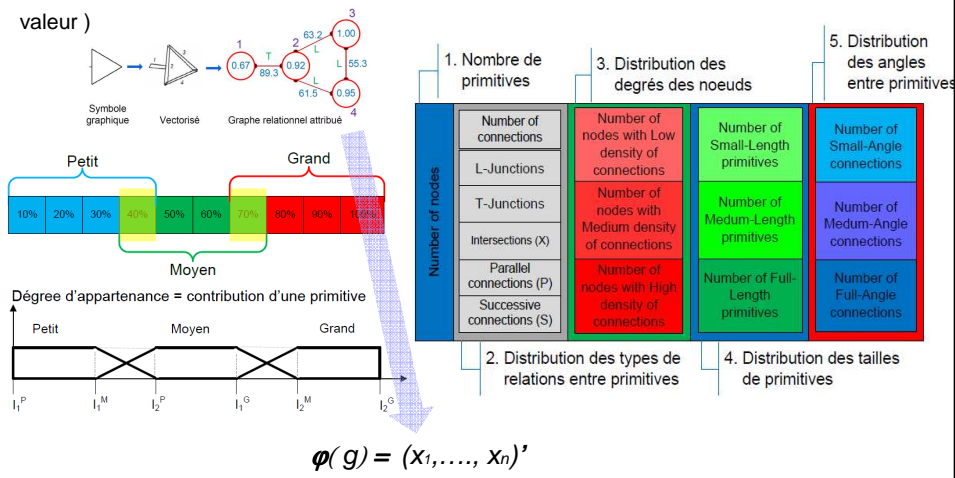
Différents types d'information peuvent être extraits du graphe

→ Construction d'un vecteur de caractéristiques statistiques

Reconnaissance des Formes à base de graphes

Graph Embedding « flou » [Luqman09]

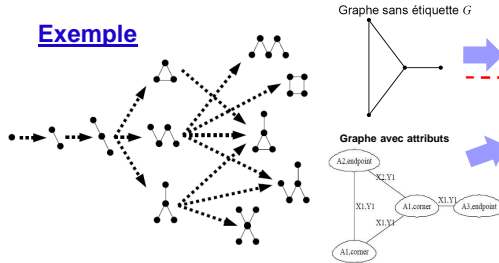
Fonction d'appartenance floue (définissant la contribution d'une primitive en fonction de sa valeur)



Reconnaissance des Formes à base de graphes

Graph Embedding [Sidère09]

Exemple



Pattern						
Freq.	4	4	5	2	1	1
A1, corner	2	4	9	4	2	2
A2, corner	0	0	0	0	0	0
A3, corner	0	0	0	0	0	0
A1, endpoint	0	0	0	0	0	0
A2, endpoint	1	2	4	2	1	1
A3, endpoint	1	1	2	2	0	1
X1, Y1	0	3	7	5	2	2
X1, Y2	0	0	0	0	0	0
X2, Y1	0	1	3	1	1	1
X2, Y2	0	0	0	0	0	0

Lexique des motifs topologiques

Fréquences d'apparition des motifs

→ Construction d'un vecteur ou matrice de caractéristiques statistiques

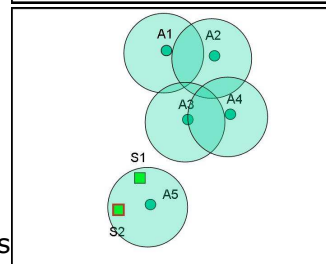
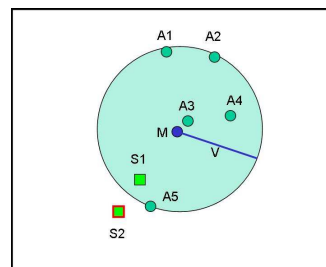
Problèmes avec le embedding, probing, kernel

Quelle transformation φ ? Quelle mesure de similarité ?

Pas d'information sur les appariements entre sommets des deux graphes

Apprentissage et Choix des modèles

- Modèle classique
 - Un seul modèle par classe fonction de la moyenne et de l'écart-type de chacune des caractéristiques des exemples de l'ensemble d'apprentissage
- Plusieurs modèles
 - Conservation de toute l'information, c'est-à-dire le vecteur de caractéristiques de chacun des exemples de l'ensemble d'apprentissage
 - But :
 - Affiner le modèle en le rendant plus proche de la réalité
 - Prendre en compte la variabilité



Apprentissage et graphes

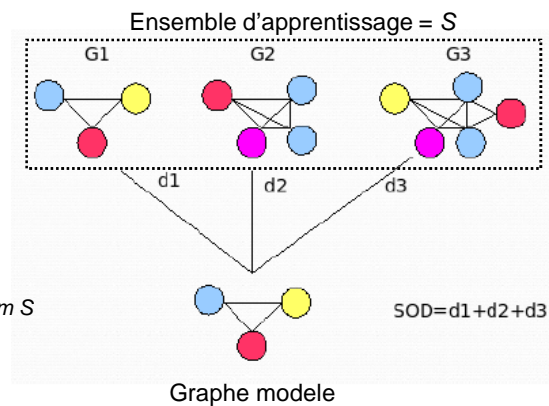
Pleins de graphes modèles ou 1 représentant ?

→ Graphe médian

$$\bar{g} = \operatorname{arg}_{g \in C} \min SOD(g, S)$$

$$SOD(g, S) = \sum_{i=1}^{|S|} d(g, g_i)$$

S = a set of graphs
 C = set of possible graphs derived from S
 D = an edit distance



71

Bilan ...

- Un pouvoir de représentativité très élevé
 - Topologie, structure, composition, ...
 - Choix du mode de représentation (nœud, arc)
 - Choix des attributs (symboliques, numériques, ...)
 - Temps de construction, taille, ...
- Traitements des graphes
 - Analyse locale → segmentation, localisation
 - Voir Recherche Opérationnelle (nombreux algorithmes ?)
- Comparaison de graphes
 - Nombreuses évolutions
 - Très forte complexité
 - Utilisation d'heuristiques
 - Transformation en vecteurs statistiques
 - Perte d'intérêt ?
- La dernière des règles de la RdF de J. C. Simon :
Surtout ne jamais désespérer !

72