



UFR des sciences
Master recherche *Génie Informatique*

Séance 1

Programmation Dynamique

Au cours de cette séance de travaux pratiques vous allez mettre en œuvre une technique de calcul de distance entre deux séquences au moyen de la programmation dynamique. La programmation se fera sous Matlab. Vous allez l'appliquer pour reconnaître des chiffres manuscrits tracés à l'aide de la souris.

1. Acquisition des données

Mettre en œuvre la fonction Acquisition donnée en annexe pour tracer des séquences de points représentant des chiffres et les sauvegarder.

2. Reconnaissance de séquences

Pour procéder à la reconnaissance de séquences vous allez calculer la distance de la séquence à reconnaître à chacune des dix séquences de référence modélisant un chiffre. Le chiffre correspondant au modèle le plus proche sera attribué à la classe de la séquence à reconnaître.

La distance entre les chaînes sera calculée par programmation dynamique. La distance entre deux points élémentaires sera la distance euclidienne.

- Visualiser sur un exemple le résultat du calcul de la distance obtenue en affichant les points des deux séquences qui sont mis en correspondance. Expliquer le résultat obtenu.
- Utiliser cette technique pour effectuer la reconnaissance de chiffres. Utiliser les tracés des autres binômes pour avoir davantage de données.
- Proposer une autre distance élémentaire qui soit moins sensible à la taille des tracés

ANNEXE

```
1)D(0,0)=0;
2)Pour i=1 à n
    faire D(i,0)=D(i-1,0)+d(ai, b1);
3)    Pour j=1 à m
        faire
            D(0,j)=D(0,j-1)+d(a1 ,bj);
4)    Pour i=1 à n                /*boucle principale
        Pour j=1 à m
            faire
                m1 = D(i-1,j-1) + 2*d(ai,bj);
                m2 = D(i-1,j)    + d(ai, bj);
                m3 = D(i,j-1)    + d(ai ,bj);
5)Prec(0,0)=0;                    /*initilisation
        D(i,j)=Min(m1 m2 m3);
        Prec(i,j)=Argmini=1,2,3(mi); /*mémorisation du meilleur chemin
6)d(X,Y)=D(n,m);
7)i=n;j=m;c=0;                    /* calcul du chemin optimal
8)tantque Prec(i,j) = 0          /* par backtrack
    faire
        chemin(c) = prec(i,j);
        c=c+1;
        selon prec(i,j)
            faire
                cas 1: i=i-1;j=j-1;
                cas 2: i=i-1;
                cas 3: j=j-1;
```

function Acquisition()

```
axis([0 10 0 10])
hold on
% Initially, the list of points is empty.
xy = [];
n = 0;
% Loop, picking up the points.
disp('Left mouse button picks points.')
disp('Right mouse button picks last point.')
but = 1;
while but == 1
    [xi,yi,but] = ginput(1);
    plot(xi,yi,'ro')
    n = n+1;
    xy(:,n) = [xi;yi];
end
% Interpolate with a spline curve and finer spacing.
t = 1:n;
ts = 1: 0.2: n;
xys = spline(t,xy,ts);
% Plot the interpolated curve.
plot(xys(1,:),xys(2:,:), 'rx');
save cinq xys;
end ;
% Chargement des données en mémoire
load cinq;
figure;
plot(xys(1,:),xys(2:,:), 'rx');
```