

Rapport sur le projet METAMATCH mois de septembre.

Introduction

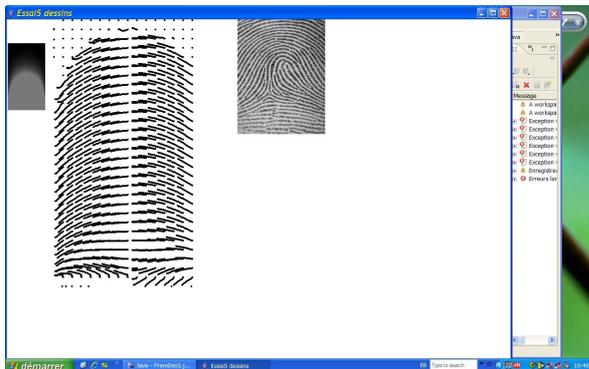
Nous avons choisi de pousser davantage la formulation du problème en termes de champs de déformation, plutôt qu'en terme de formations de lignes par un formalisme de type cristal liquide (voire précédents rapports). Il apparaît que le formalisme en termes de champs de déformation décrit le phénomène « naturel » de la formation d'une empreinte digitale. Le travail se poursuit donc en étudiant les différents problèmes liés au formalisme de champ de déformation. L'objectif sera de trouver le formalisme biomécanique permettant de former des empreintes réelles.

Principe des calculs.

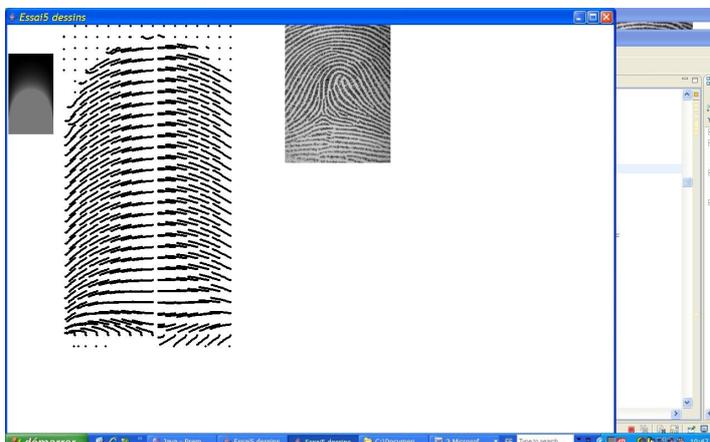
Les différents champs de lignes obtenus sont les lignes tangentes au vecteur du repère principal courbure d'une membrane fléchie sous l'effet d'une distribution de forces.

Voici des exemples de sorties obtenues, par une étude de quelques fonctions types de « foca » ou de « charge » déposée sur la membrane.

Charge le long de la phalangette $= -20$ aucune autre charge sur la membrane

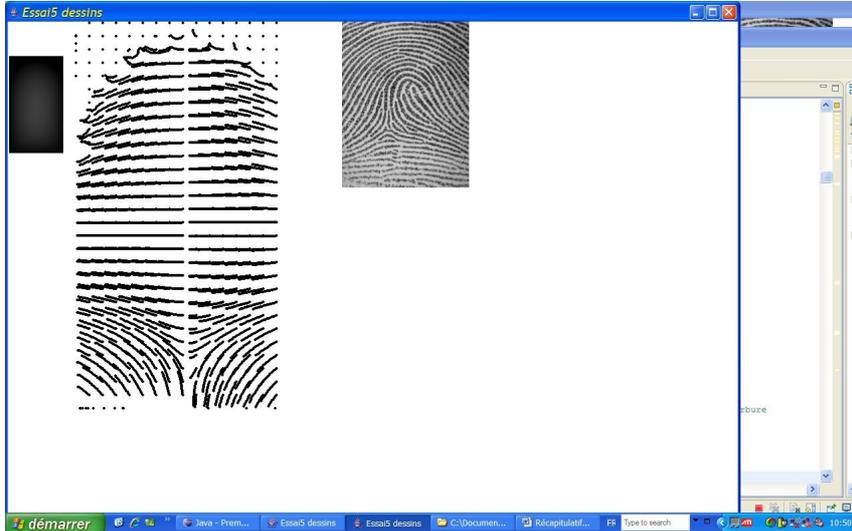


Charge le long de la phalangette $= -2$ aucune autre charge sur la membrane

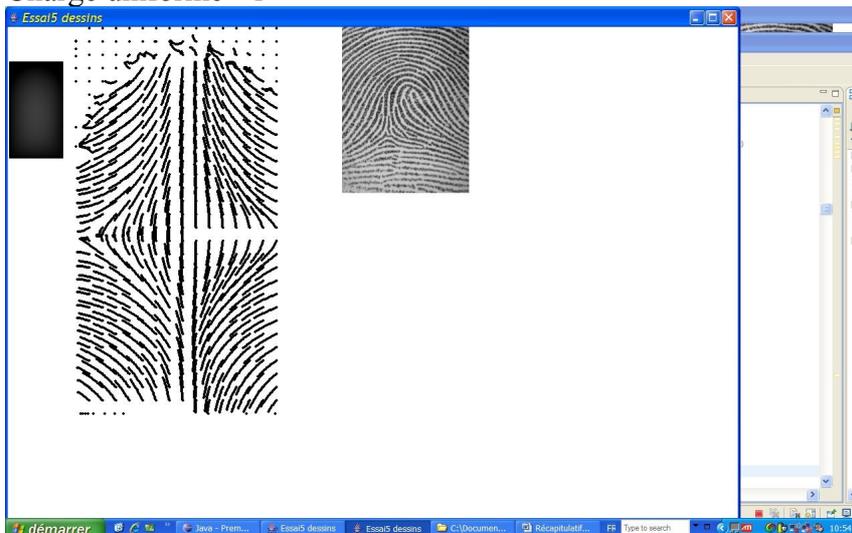


C'est identique=>linéarité en charge

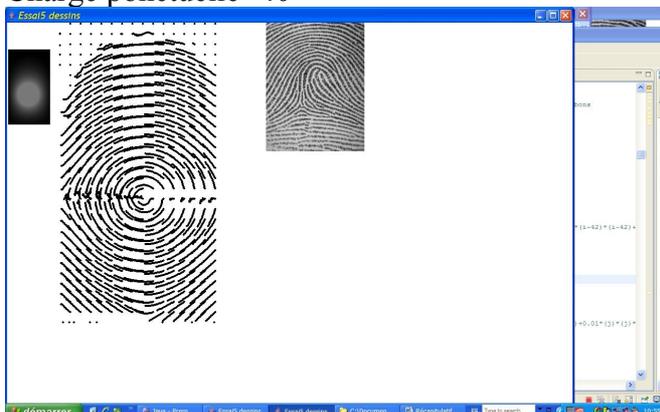
Charge uniforme -1



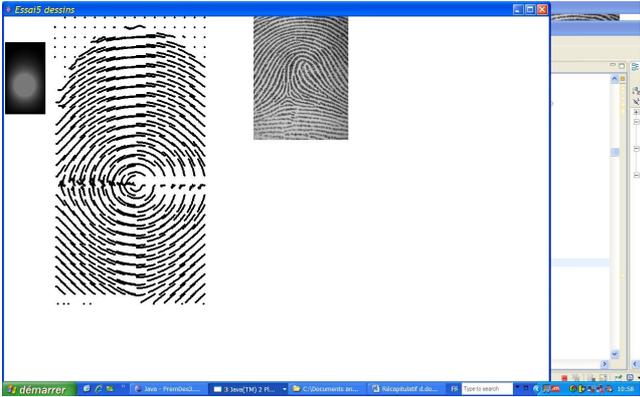
Charge uniforme +1



Charge ponctuelle -40

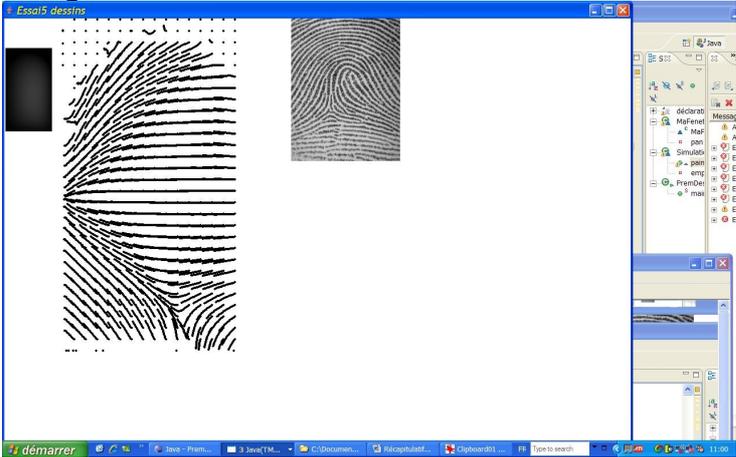


Charge ponctuelle -4

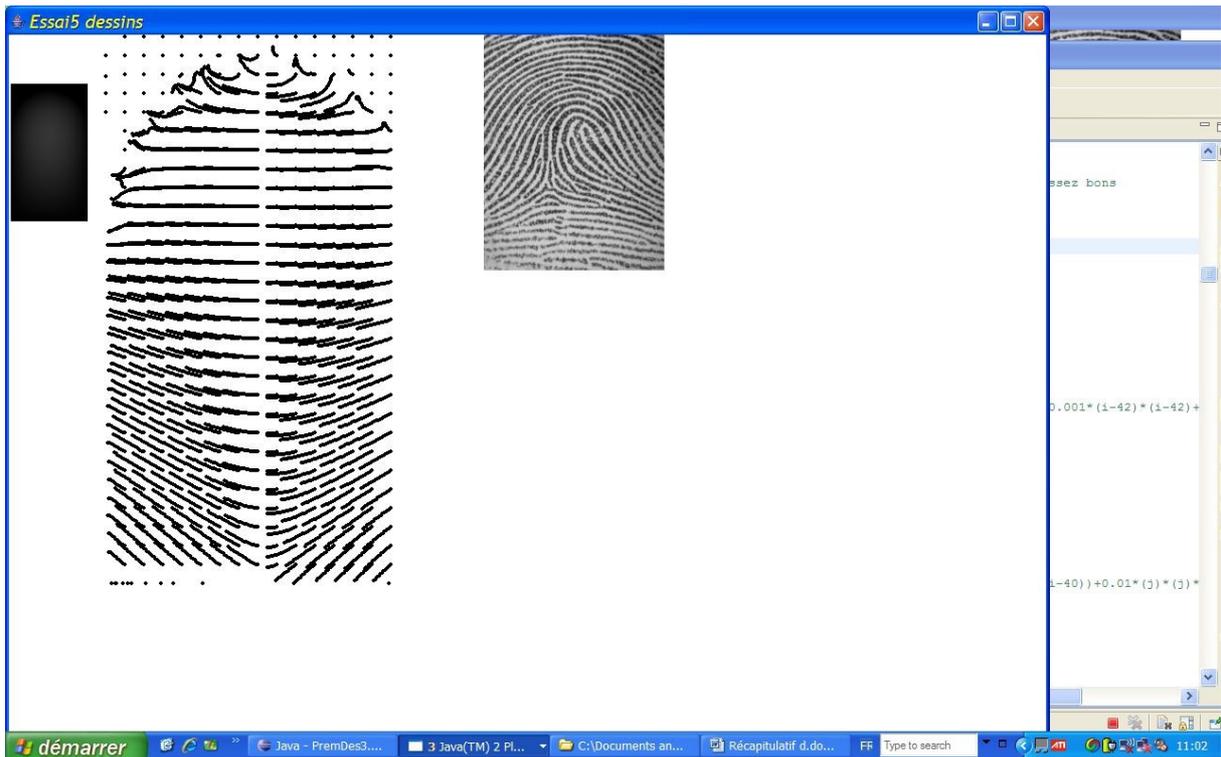


Idem -40=> linéarité

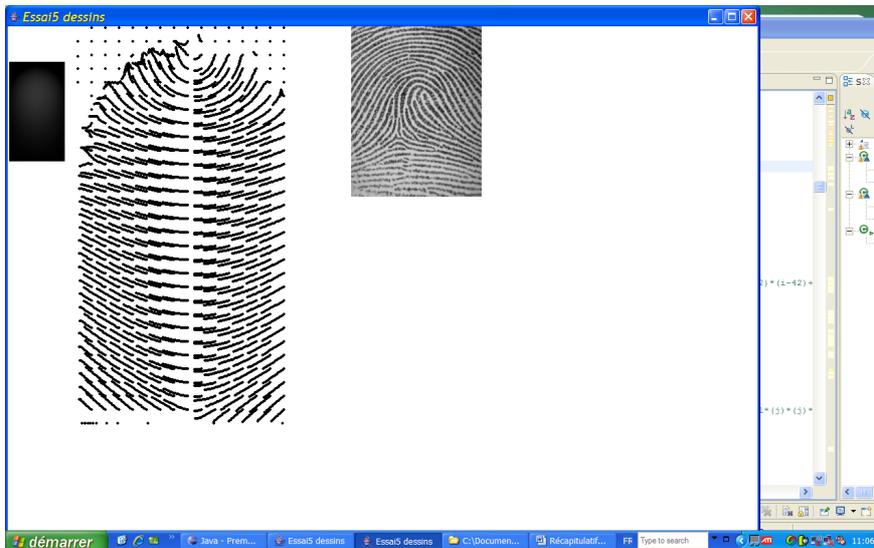
Charge linéaire -X



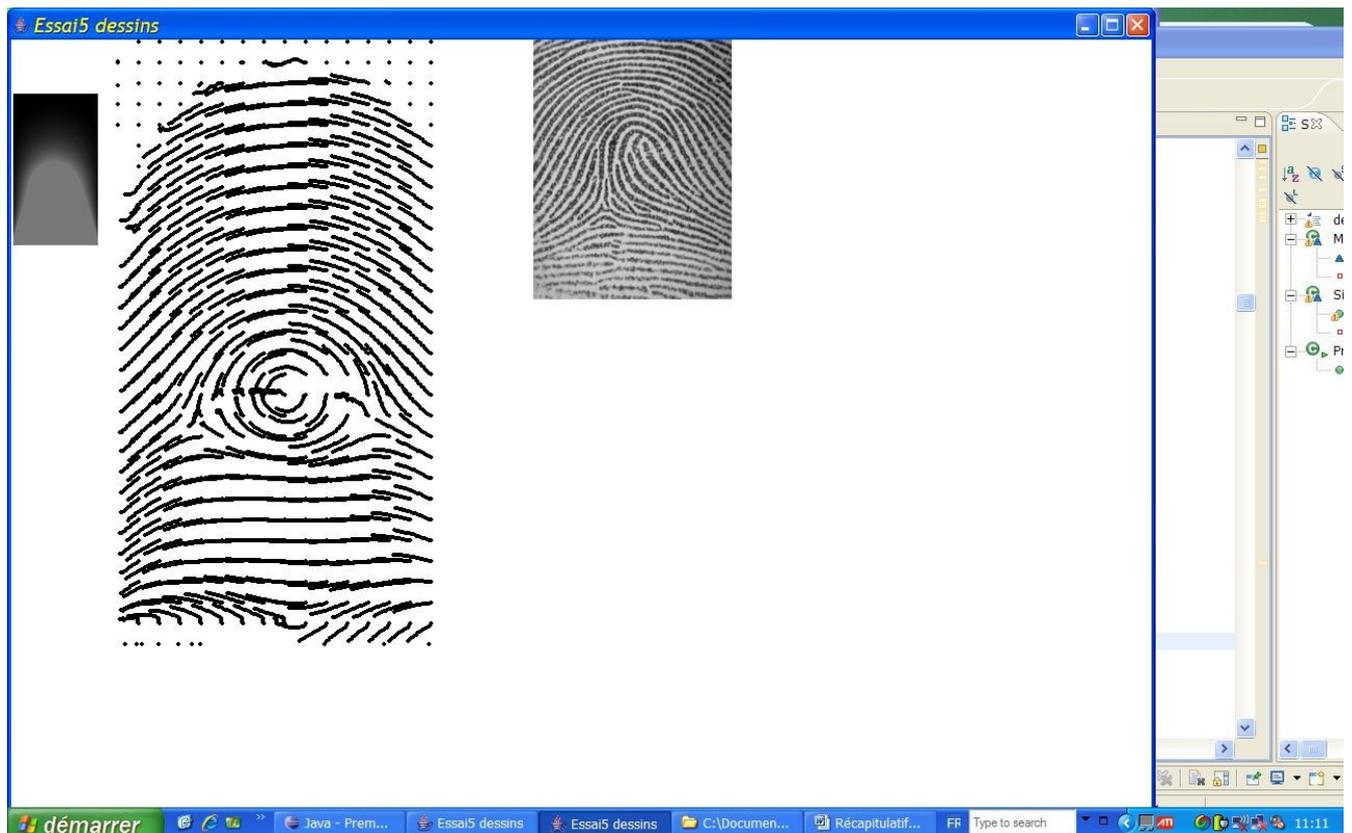
Charge linéaire -Y



Charge $-Y^2$



Phalange -1 plus charge ponctuelle -4



On découvre peu à peu une richesse du phénomène, et un espace de formes assez variées. Il existe en fait beaucoup plus de formes possibles, a priori, que simplement les empreintes digitales. Il est probable que les empreintes digitales réelles sont très limitées par le mécanisme de formation de plis à la surface d'un doigt, la forme-même des doigts étant assez reproductible (aux variations inter-individus près).

Réflexion sur le tracé des champs d'empreintes

La visualisation des lignes générées par le code est un élément important pour l'appréciation du résultat.

Nous avons implémenté deux types de visualisation. La première consiste à partir d'une succession de points situés à intervalle réguliers et à suivre leur mouvement en intégrant l'équation

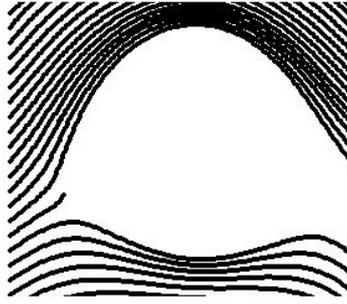
$$dM/dt=n$$

$$dxm/dt=nx$$

$$dym/dt=ny$$

Le code pour cela est donné en annexe.

Ce type de visualisation engendre des lignes à l'aspect suivant :



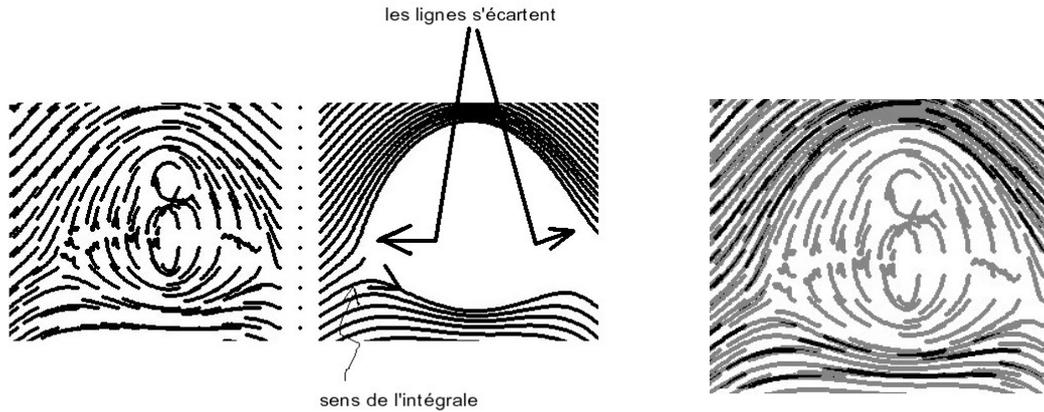
Un inconvénient est que si les points initiaux sont sur le bord, on manque les domaines clos contournés par les lignes. On peut donc faire partir les lignes du centre pour « remplir » les endroits qui sont contournés.

Cependant, malgré le réalisme apparent ces lignes ne sont pas l'équivalent des lignes d'empreintes digitales. Ce sont des trajectoires qui suivent le champ de vecteur \mathbf{n} engendré par le calcul. En effet, au cours du déplacement dans le champ de vecteur, a trajectoire ne constitue pas à proprement parler une ligne de courant de \mathbf{n} .

C'est pourquoi nous avons choisi de montrer également une autre représentation de \mathbf{n} qui consiste à tracer en chaque point (i,j) un « brin » obtenu en intégrant sur quelques pas de temps le champ de vecteur $\mathbf{n} (n_x, n_y)$. On obtient alors une représentation de la forme :



Cette représentation n'est pas équivalente, comme le montre la comparaison des deux dessins :



La région des flèches « les lignes s'écartent », montre que au fil de la trajectoire, l'intégrale du mouvement « le long » de \mathbf{n} induit un écart de la ligne tracée par rapport à la ligne de courant enveloppe du champ \mathbf{n} . L'image de droite montre une projection de la représentation en brins sur la représentation intégrale. Il faut donc garder à l'esprit cette nuance, quand on regarde les images.

Réflexion sur le problème inverse

A terme, il faudra traiter le problème inverse C'est-à-dire : connaissant le champ \mathbf{n} , reconstruire une charge G telle que

$\Delta(G)$ =charge et

\mathbf{n} =lignes de courbure principale de G

Soit $G_{xx}=d^2G/dx^2$
 $G_{yy}=d^2G/dy^2$

On a :

$\Delta(G)=G_{xx}+G_{yy}=\text{Charge}(x,y)$

Et

$\text{alfa}=(\text{lambda}-G_{xx})/G_{xy}$

Lambda est la valeur propre locale. On va prendre lambda=1 (tension constante ?)

$n_x=1/(1+\text{alfa}*\text{alfa})^{1/2}$;

$n_y=\text{alfa}/(1+\text{alfa}*\text{alfa})^{1/2}$;

$$n_x = 1 / (1 + ((\lambda - G_{xx}) / G_{xy})^2)^{1/2};$$

$$n_y = (\lambda - G_{xx}) / G_{xy} / (1 + ((\lambda - G_{xx}) / G_{xy})^2)^{1/2};$$

$$(n_y^2 / n_x^2) = (\lambda - G_{xx}) / G_{xy};$$

$$n_y^2 = (\lambda - G_{xx}) / G_{xy} / (1 + ((\lambda - G_{xx}) / G_{xy})^2);$$

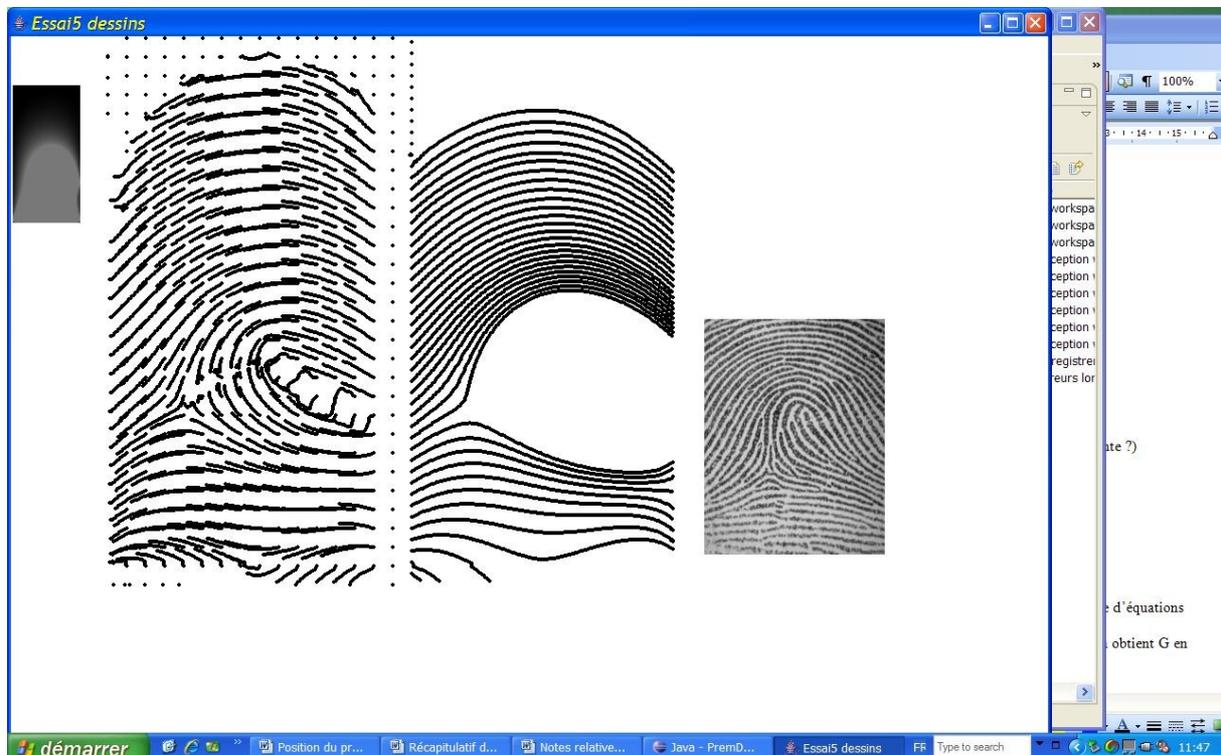
ceci produit en principe G_{xx} et G_{yy} , connaissant en tout point n_x et n_y , à condition de savoir inverser ce système d'équations. Le moment venu, il faudra réfléchir à un algorithme de résolution de ces équations.

Réflexions sur les sorties.

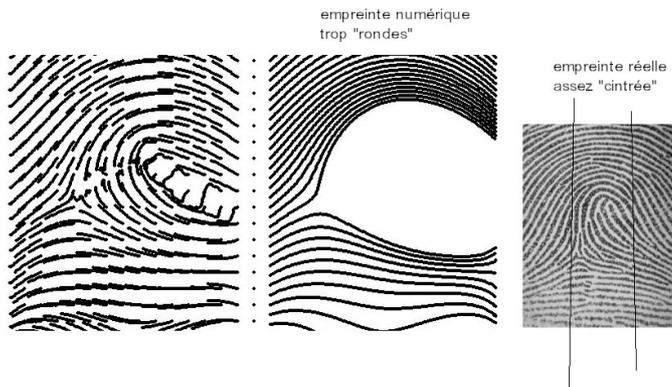
Il apparaît assez facile de former des cibles ou des tentes, la formation de boucles paraît plus difficile. Les travaux de Kücken et al. (voir précédents rapport) attribuent la formation de boucles à un biais de la surface du doigt.

Nous avons essayé de former des boucles en forçant une déformation « en boucle » de la surface de la membrane.

Ceci s'obtient en implémentant une charge qui est déposée le long d'une ligne formant boucle. (voir code en annexe)



L'impression globale d'un grand nombre d'essais est que les empreintes obtenues sont trop « rondes ».



Ceci sera étudié au mois d'octobre.

Programme d'étude pour le mois d'octobre : essais de « cintrage d'empreintes en forme de boucles

Annexe

Code permettant de définir une charge en forme de boucle

```
//definition du filament pour la boucle  
//le filament est défini par son centre et la direction de pencher
```

```
abscisse=1;  
centreboucler=Centrex;  
centreboucley=Centrey;  
poidsaucentre=0.2;
```

```
j=(int)centreboucley;
```

```
pentex=+0;  
pentey=-1.5;  
courbx=+0.04;  
courbx3=-0.0001;  
courby=-0.0;  
poids=0.1;
```

```
//la boucle est une courbe du troisième degré partant du point central centreboucle
```

```
for(abscisse=1;abscisse<65;abscisse++)  
{boucler[abscisse]  
=centreboucler+pentex*abscisse+courbx*abscisse*abscisse  
+courbx3*abscisse*abscisse*abscisse;
```

```
boucley[abscisse]
    =centreboucley+pentey*abscisse+courby*abscisse*abscisse;
}
```

```
for(abscisse=1;abscisse<65;abscisse++)
```

```
{
```

```
    i=(int)(bouclex[abscisse]);
    j=(int)(boucley[abscisse]);
    if ( ( (i)<maxx-1)
          & ( i)>0 )
        & ( (j)<maxy-1)
          & ( j)>0 )
```

```
    {
    charge[i][j]=
```

```
    charge[i][j]-poids*(1+0.6*abscisse);
```

```
    //-poids;
```

```
    //charge[i][j]-poids;
```

```
    }
```

```
}
```

//cette charge est ensuite utilisée comme condition de charge pour le calcul de la déformation
(voir précédents rapports)