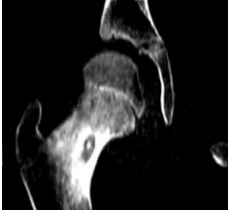
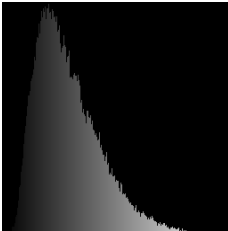

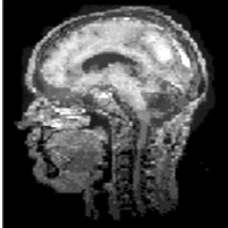


TP de Traitement d'Images Médicales : Exemple d'implémentation

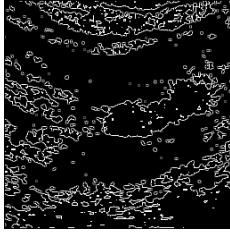
Gwenaël Brunet, Laurent Lecornu

Département Image et Traitement de l'Information - GET/ENST de Bretagne

Gwenael.Brunet@enst-bretagne.fr, Laurent.Lecornu@enst-bretagne.fr

Méthode	Implémentation
Contraste 	<pre>for(y=0; y<HAUTEUR; y++) { for(x=0; x<LARGEUR; x++) { fpixel = ((float)image[x][y]-fmax)*5.0f+fmax; fpixel = fpixel>255?255:fpixel; fpixel = fpixel<0?0:fpixel; imageTemp[x][y] = (unsigned char)fpixel; } }</pre>
Histogramme 	<pre>// Initialisation de l'histogramme à "0" for(i=0; i<256; i++) histogramme[i] = 0; // Initialisation du rendu de l'histogramme à "0" for(y=0; y<HAUTEUR; y++) for(x=0; x<LARGEUR; x++) imageT[x][y] = 0; // Calcul de l'histogramme int max = 0; for(y=0; y<HAUTEUR; y++) { for(x=0; x<LARGEUR; x++) { histogramme[image[x][y]]++; if(max<histogramme[image[x][y]]) max = histogramme[image[x][y]]; } } // Rendu de l'histogramme for(x=0; x<LARGEUR; x++) { for(y=HAUTEUR-1; y>HAUTEUR-((float)histogramme[x]/max*255); y--) imageT[x][y] = x; } }</pre>
Fusion intra-modalité 	<pre>for(y=0; y<HAUTEUR; y++) { for(x=0; x<LARGEUR; x++) { pixel = image[x][y]-image2[x][y]; if(pixel<0) pixel = 0; imageT[x][y] = pixel; } }</pre>
Fusion inter-modalité 	<pre>for(y=0; y<HAUTEUR; y++) { for(x=0; x<LARGEUR; x++) { pixel = image[x][y]+(imageC[x*3][y]+imageC[x*3+1][y]+ imageC[x*3+2][y])/3; if(pixel>255) pixel = 255; imageT[x][y] = pixel; } }</pre>

Segmentation



```
// Réalisation de l'histogramme afin de connaître le niveau de gris  
// optimal pour le seuillage (fmax)...
```

```
int max = 0;  
float fmax = -1;  
for( y=0; y<HAUTEUR; y++ )  
{  
    for( x=0; x<LARGEUR; x++ )  
    {  
        histogramme[image[x][y]]++;  
        if( max<histogramme[image[x][y]] )  
        {  
            max = histogramme[image[x][y]];  
            fmax = image[x][y];  
        }  
    }  
}  
  
// Traitement seuillage  
for( y=0; y<HAUTEUR; y++ )  
{  
    for( x=0; x<LARGEUR; x++ )  
        imageTemp[x][y] = image[x][y]>fmax?255:0;  
}
```

```
// Contours Laplacien
```

```
for( y=1; y<HAUTEUR-1; y++ )  
{  
    for( x=1; x<LARGEUR-1; x++ )  
    {  
        pixel = (imageTemp[x-1][y-1]*-1 + imageTemp[x][y-1]*-1 +  
                imageTemp[x+1][y-1]*-1 + imageTemp[x-1][y]*-1 +  
                imageTemp[x][y]*8 + imageTemp[x+1][y]*-1 +  
                imageTemp[x-1][y+1]*-1 + imageTemp[x][y+1]*-1 +  
                imageTemp[x+1][y+1]*-1)/3;  
        pixel = pixel>255?255:pixel;  
        pixel = pixel<0?0:pixel;  
        imageT[x][y] = pixel;  
    }  
}
```

Ouverture



```
int num = MEDIAN/2;  
for( y=num; y<HAUTEUR-num; y++ )  
{  
    // Erosion  
    for( x=num; x<LARGEUR-num; x++ )  
    {  
        n=0;  
        for( j=y-num; j<y+num+1; j++ )  
            for( i=x-num; i<x+num+1; i++ )  
                median[n++] = image[i][j];  
  
        qsort( median, MEDIAN*MEDIAN, sizeof(unsigned char), compare );  
        imageTemp[x][y] = median[0];  
    }  
}  
for( y=num; y<HAUTEUR-num; y++ )  
{  
    // Dilatation  
    for( x=num; x<LARGEUR-num; x++ )  
    {  
        n=0;  
        for( j=y-num; j<y+num+1; j++ )  
            for( i=x-num; i<x+num+1; i++ )  
                median[n++] = imageTemp[i][j];  
        qsort( median, MEDIAN*MEDIAN, sizeof(unsigned char), compare );  
        imageT[x][y] = median[MEDIAN*MEDIAN-1];  
    }  
}
```