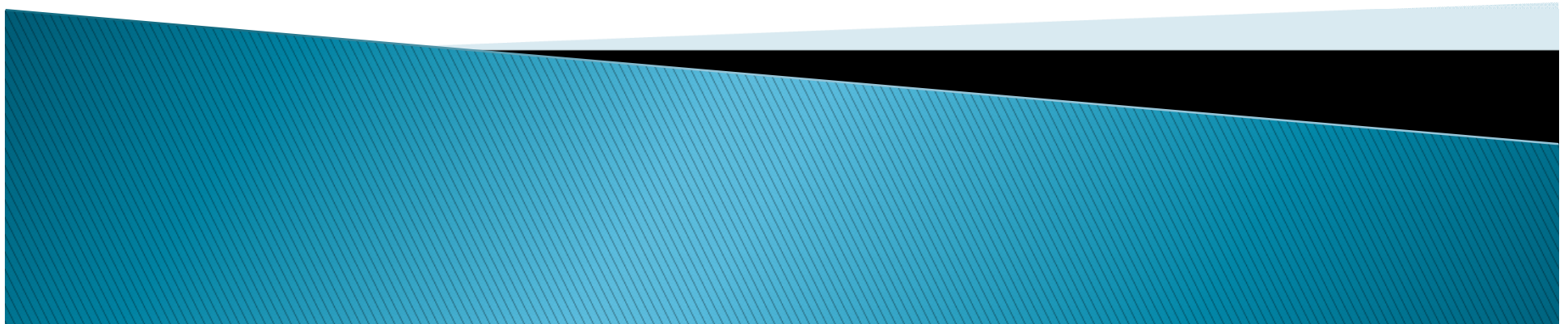


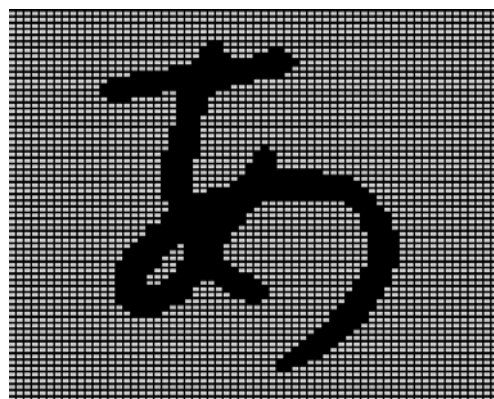
# 知識工学 I 課題4

東京工業高等専門学校 情報工学科  
鈴木雅人



# 手書き文字認識の概略

入力画像



特徴抽出

特徴量

識別

認識  
結果

比較

前処理

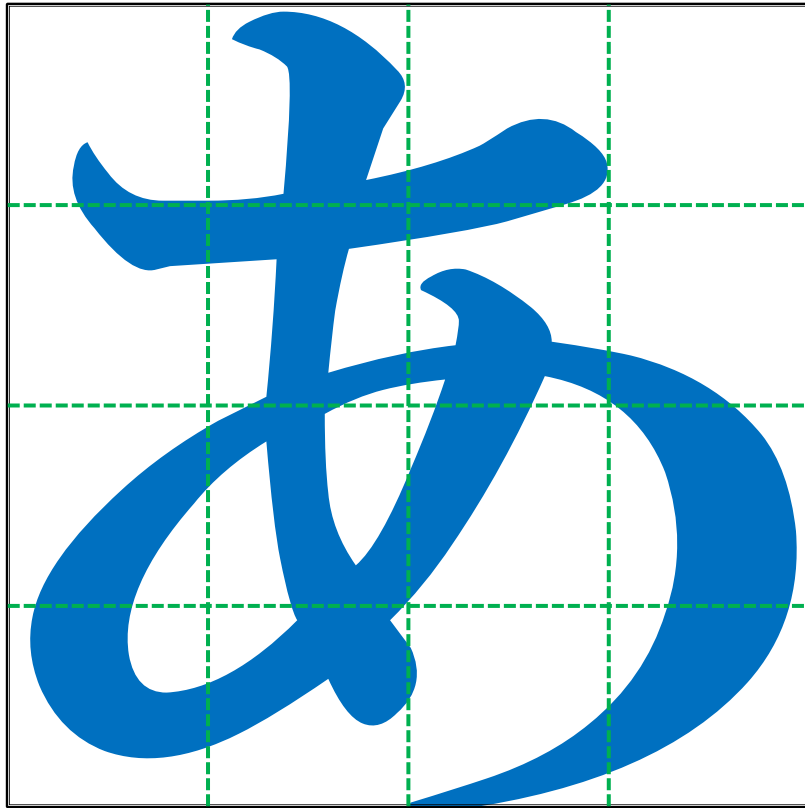
ノイズ除去  
スムージング  
正規化  
輪郭抽出ほか  
...  
特徴抽出

字形の多様性を吸収

あ...(15,23,82,9,...)  
い...(29, 8,73,41,...)  
う...(0,21,43,102,...)  
...  
...  
ん...(19,94,26,43,...)

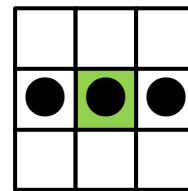
辞書(標準パターン)

# 特徴量の抽出例

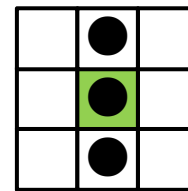


(1) 小領域ごとに黒画素数を数える  
→ 16次元の特徴量

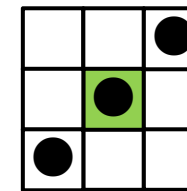
(2) 小領域ごとに4方向線素数を数える  
→  $16 \times 4 = 64$ 次元の特徴量



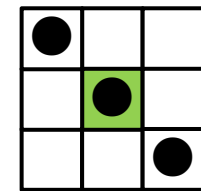
横線素



縦線素



右上り  
線素

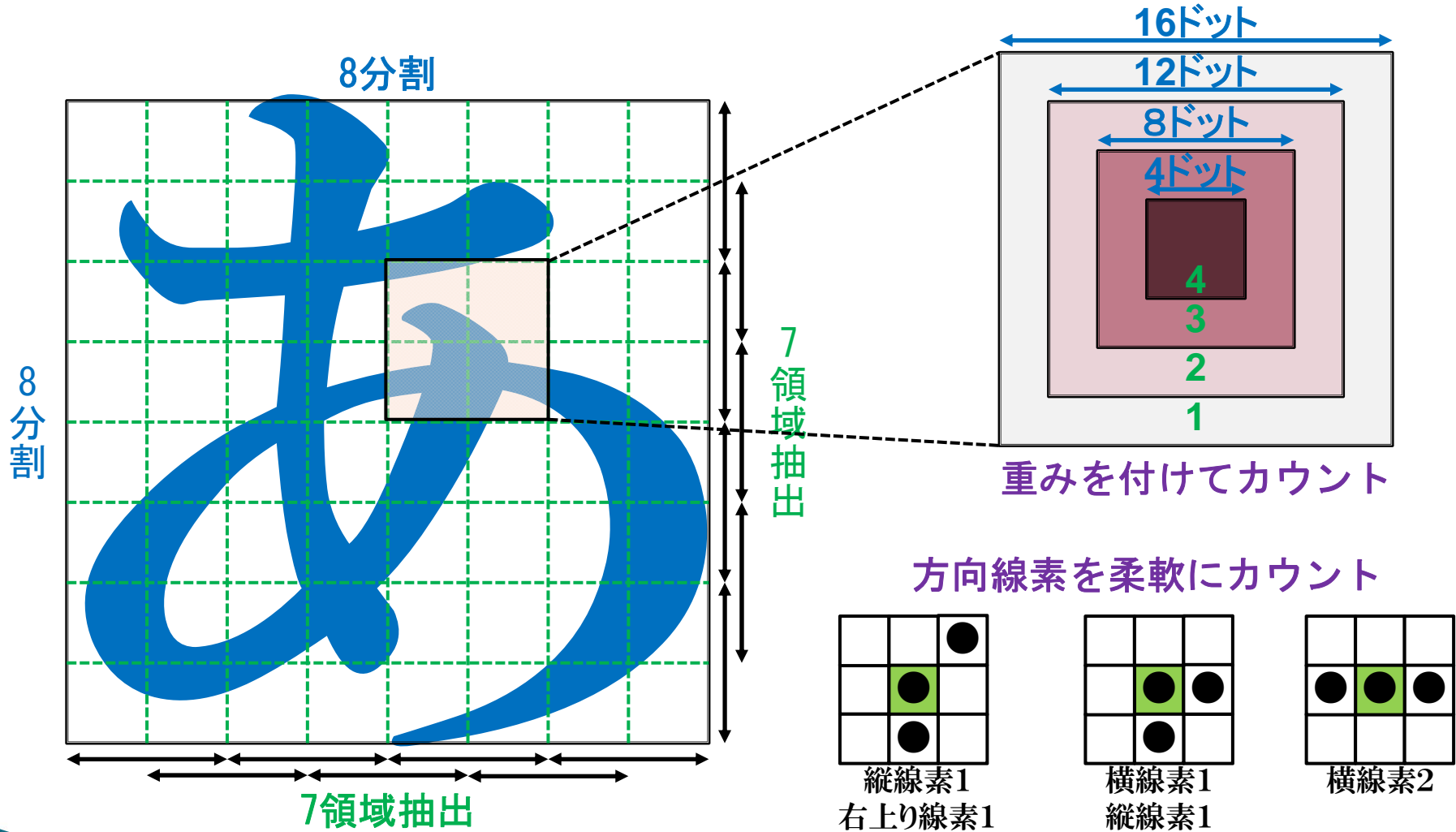


左上り  
線素

## 【注意】

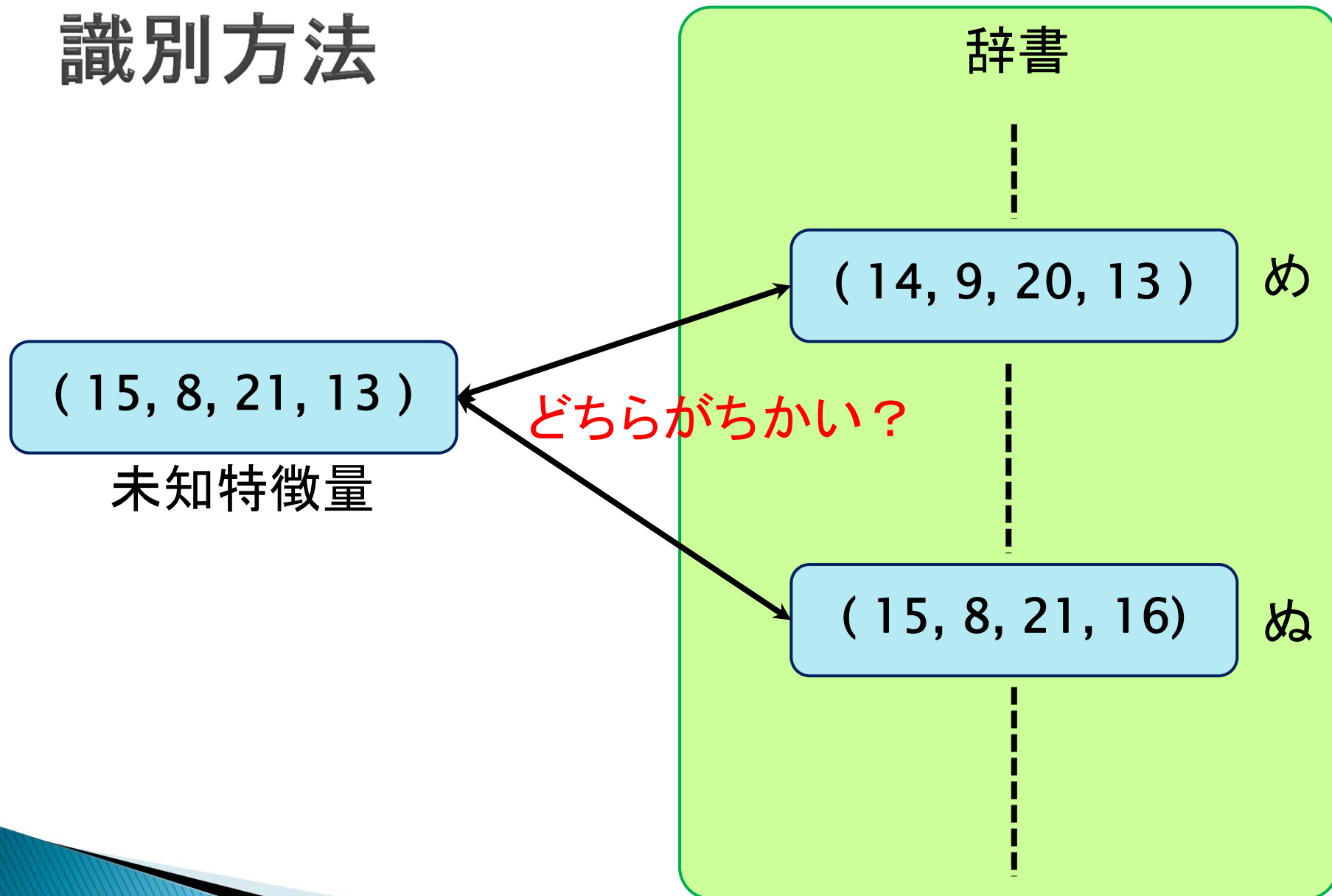
- ◆ 小領域の数や区切り方には様々なパターンがある.
- ◆ 線素は縦・横・右上り斜め・左上り斜めの4方向が一般的. 縦・横のみを考える場合もある.
- ◆ 線分などの大局的特徴は正確に取れないことが多い.

## 【参考】方向線素特徴量



**特徴量の次元**  $7 \times 7$  (領域)  $\times 4$  (方向) = 196次元

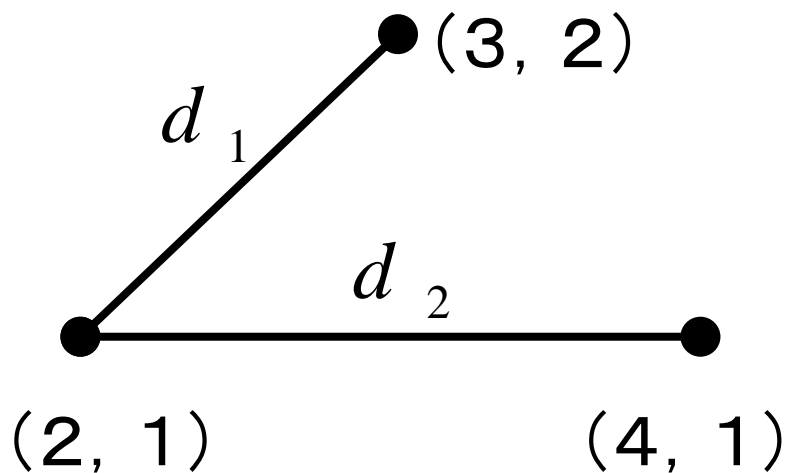
# 識別方法



# 識別方法

$$d_1 = \sqrt{(3 - 2)^2 + (2 - 1)^2} = \sqrt{2}$$

$$d_2 = \sqrt{(4 - 2)^2 + (1 - 1)^2} = 2$$



# 識別方法

( 15, 8, 21, 13 )

未知特徴量

どちらがちかい？

$d_1$

$d_2$

( 14, 9, 20, 13 )

め

( 15, 8, 21, 16 )

ぬ

$$\begin{aligned} d_1 &= \sqrt{(15 - 14)^2 + (8 - 9)^2} \\ &\quad + (21 - 20)^2 + (13 - 13)^2 \\ &= \sqrt{3} \end{aligned}$$

同様に  $d_2 = 3$

# 距離の三公理

任意のベクトルに対し、次の式を全て満たす $d$ を距離という.

非負性

$$d(\vec{x}, \vec{y}) \geq 0$$

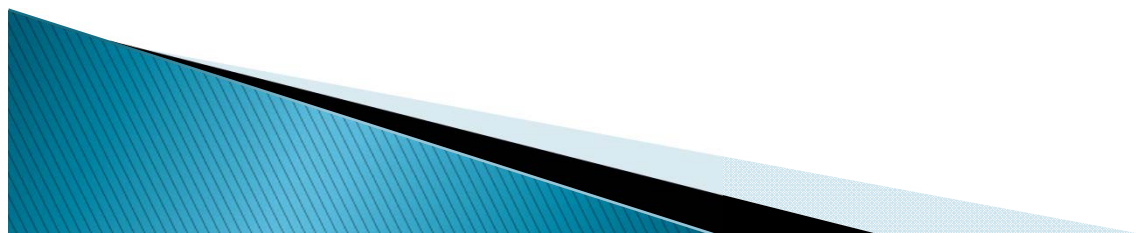
$$\vec{x} = \vec{y} \text{ のときのみ} \quad d(\vec{x}, \vec{x}) = 0$$

対称性

$$d(\vec{x}, \vec{y}) = d(\vec{y}, \vec{x})$$

三角不等式

$$d(\vec{x}, \vec{z}) \leq d(\vec{x}, \vec{y}) + d(\vec{y}, \vec{z})$$





# 課題4－1 認識手法

考案する認識手法をレポートにまとめなさい。レポートには次の内容は必ず入れること。

- ▶ 課題の内容
- ▶ 考案した認識手法（入力から出力までを明確に）
- ▶ これまで作成した関数の活用法



## 課題4-2（発展課題）

下記の課題について調査し、レポートにまとめなさい。

- ▶ ユークリッド距離は距離の三公理を満たすことを示しなさい。
- ▶ その他に距離の三公理を満たすものを調べなさい。
- ▶ 俗に「〇〇距離」という名前がついているにもかかわらず距離の三公理を満たさないものがある。そのような距離を探して定義式を示し、距離の三公理を満たさない理由をまとめなさい。



## 課題4－3 認識プログラムの作成

課題4－1で考案した認識プログラムを作成するため、与えられた文字画像から、その特徴量を抽出するプログラムを作成しなさい。

作成にあたっては下記のことには注意すること。

- ① 入力となるデータファイル内に文字画像は何個含まれていても正しく処理できるようにすること。
- ② 抽出した特徴量は、入力ファイルと同じファイル名で、拡張子が「ftr」のファイルに出力すること。


（例）入力が「01.img」なら出力は「01.ftr」

- ① 可能であれば、複数のデータファイルを与えても処理できるようにしておくといい。



# 【補足1】

```
int main( int argc, char *argv[] )
{
    int i, j ;
    char fn[0x80] ;
    for( i = 1 ; i < argc ; i++ ) {
        strcpy( fn, argv[i] ) ;
        for( j = strlen(fn)-1 ; j >= 0 ; j-- ) {
            if( fn[j] == '.' ) {
                fn[j]= '¥0' ;
                break ;
            }
        }
        strcat( fn, ".ftr" ) ;
        /* この後fnをオープンして処理を続ける */
    }
```



## 【補足2】

ファイル内のデータ数を調べる

```
FILE *fp ;
```

```
int num ;
```

```
fp = fopen( “ファイル名”, “rb” ) ;
```

```
fseek( fp, 0, SEEK_END ) ;
```

```
num = ftell( fp ) / 512 ;
```

```
fseek( fp, 0, SEEK_SET ) ;
```

★ftellはファイルサイズ（バイト数）を返す

★fseekはファイル内の参照位置を変更するために使う

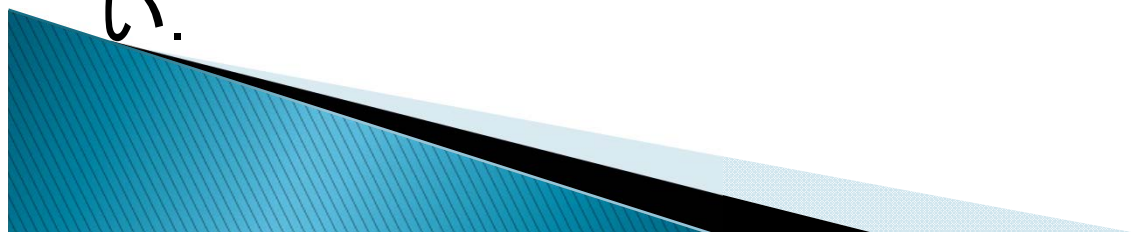


## 課題4-4 辞書生成プログラムの作成

- ① 課題4-3で作成したプログラムを用いて、サンプルデータ01.img～46.imgを処理し、01.ftr～46.ftrを生成する。
- ② 01.ftr～46.ftrを読み込み、字種ごとの平均特徴量を計算し、それをave.dicに保存するプログラムを作成しなさい。

### 【注意】

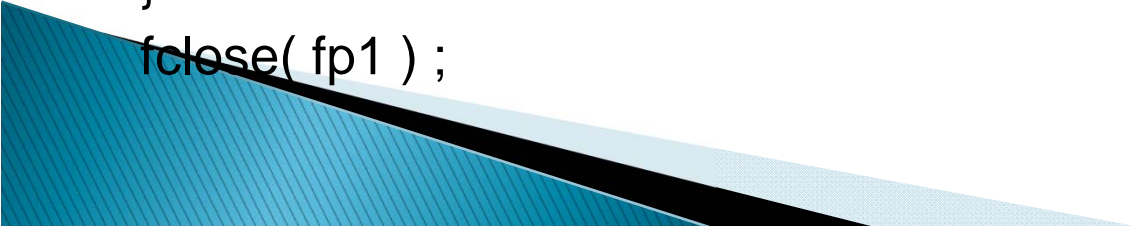
ave.dicの中身はどのような形式でも構わない。必要なら、字種ごとに平均特徴量を保存するファイルを作成しても良い。



## 【補足3】

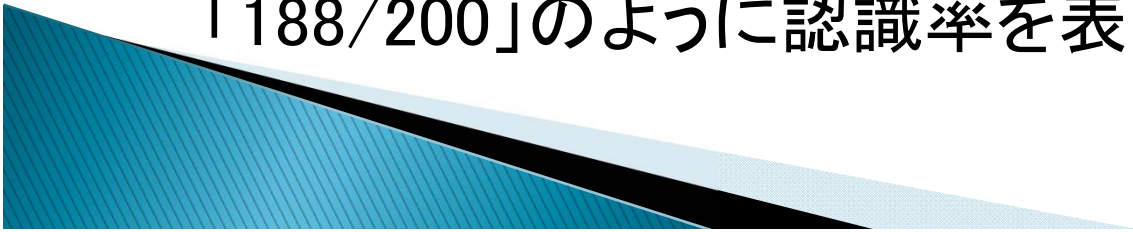
```
FILE *fp1, *fp2 ;
char fn[0x80] ;
int  data[N][Dim], i ; /* Nはサンプル数 , Dimは特徴の次元 */
Double ave[Dim] ;

fp1 = fopen( "dic.txt", "w" ) ;
for( i = 0 ; i < 46 ; i++ ) {
    sprintf( fn, "c%02.ftr", i+1 ) ;
    fp2 = fopen( fn, "r" ) ;
    /* ファイルの内容をdata[][]に読み込む */
    /* 各成分の平均ave[]を求めてfp1に書き込む */
    fclose( fp2 ) ;
}
fclose( fp1 ) ;
```



## 課題4－5 認識プログラムの作成

課題4－4で作成したave.dicを用いて，課題4－3で作成した01.ftr～46.ftrを認識し，認識結果および認識率を表示するプログラムを下記の要領で作りなさい．

- ① 認識する特徴ファイルは，同時に2つ以上与えることはせず，1つだけ与えるものとする．
  - ② 認識する特徴ファイルには，同じ字種の特徴量だけが含まれているものとする．ただしデータ数は未知とする．
  - ③ 認識結果の出力は0～45の数値でも構わない．
  - ④ 例えば，200文字中188文字を正しく認識した場合には，「188/200」のように認識率を表示すること．
- 



## 課題4－6(発展課題)

認識対象特徴量と辞書データ(標準特徴量)との距離を計算するとき、特徴量の各成分のばらつきは異なるため、ユークリッド距離による認識には問題がある。この問題を解決するには、特徴量の各成分の標準偏差(分散の平方根)を考慮すればよい。このことを用いて課題4－5のプログラムを改良しなさい。

$$D = \sqrt{\sum_{k=1}^d \left( \frac{x_k - m_k}{\sigma_k} \right)^2}$$

$\vec{x} = (x_1, x_2, \dots, x_d)$  未知の特徴量

$\vec{m} = (m_1, m_2, \dots, m_d)$  ある字種の標準特徴量

$\sigma_k$  ある字種の第 $k$ 成分に関する標準偏差