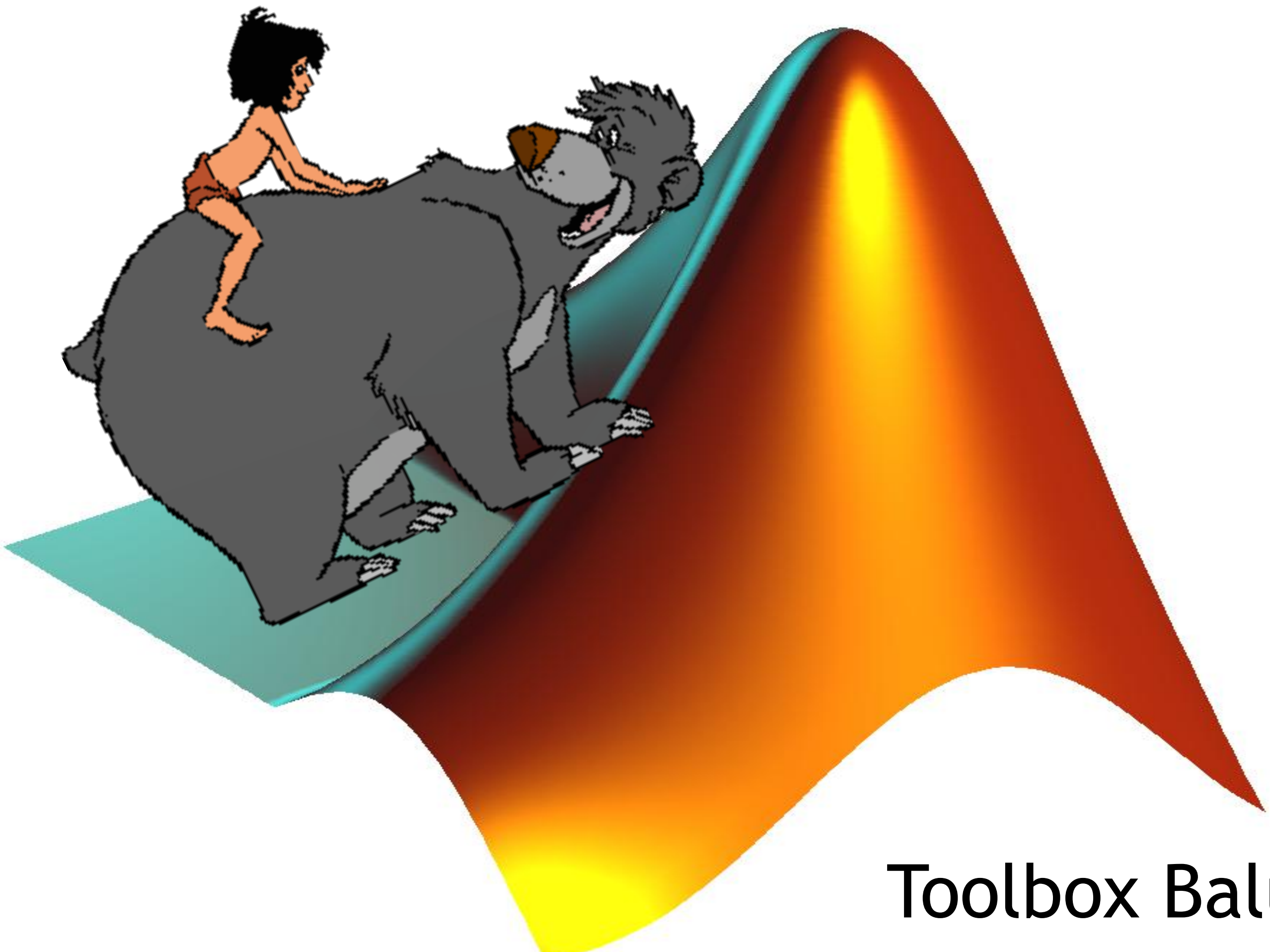


Toolbox Balu

http://dmery.ing.puc.cl/index.php/balu/

Tutorial: Toolbox Balu

Domingo Mery
Machine Intelligence Group
Department of Computer Science
Universidad Católica de Chile
http://dmery.ing.puc.cl

Toolbox Balu

More than 200 functions for image processing, feature extraction, feature transformation, feature analysis, feature selection, data selection and generation, classification, clustering, performance evaluation, multiple-view analysis, image sequence processing and tracking with geometrical constraints, see examples.

http://dmery.ing.puc.cl/index.php/balu/

Collaboration

Many improvements and codes were developed by:

Sandipan Banerjee
Patricio Cordero
Esteban Cortázar
Rodrigo González
James Kapaldo
German Larrain
Gabriel Leiva
Daniel Maturana
Carlos Mena
Cristobal Moenne
Germán Mondragón
Diego Patiño
Christian Pieringer
Vladimir Riffo
Alvaro Soto
Gabriel Tejeda
Irene Zuccar

If you want to collaborate please send me an e-mail

Copyright

NOTES:

Certain Balu functions use commands of the followings toolboxes: **VLFeat**, **Image Processing**, **Bioinformatics**, and **Neural Netowrks**. It is necessary to install this toolboxes if you want to use these Balu functions.

Certain neural network functions were implemented based on **NetLab Toolbox**: (c) 1996-2001, Ian T. Nabney, All rights reserved. Nabney, I.T. (2003): Netlab: Algorithms for Pattern Recognition, Advances in Pattern Recognition, Springer.

Certain Local Binary Patterns functions were implemented based on code written by Heikkila & Ahonen (see **http://www.cse.oulu.fi/MVG/Research/LBP**) all rights reserved.

Partial Least Squares Regression was implemented based on code developed by Gelady
(see **http://www.cdpcenter.org/files/plsr**).

Welcome to Balu Toolbox Matlab

for computer vision, pattern recognition and image processing...

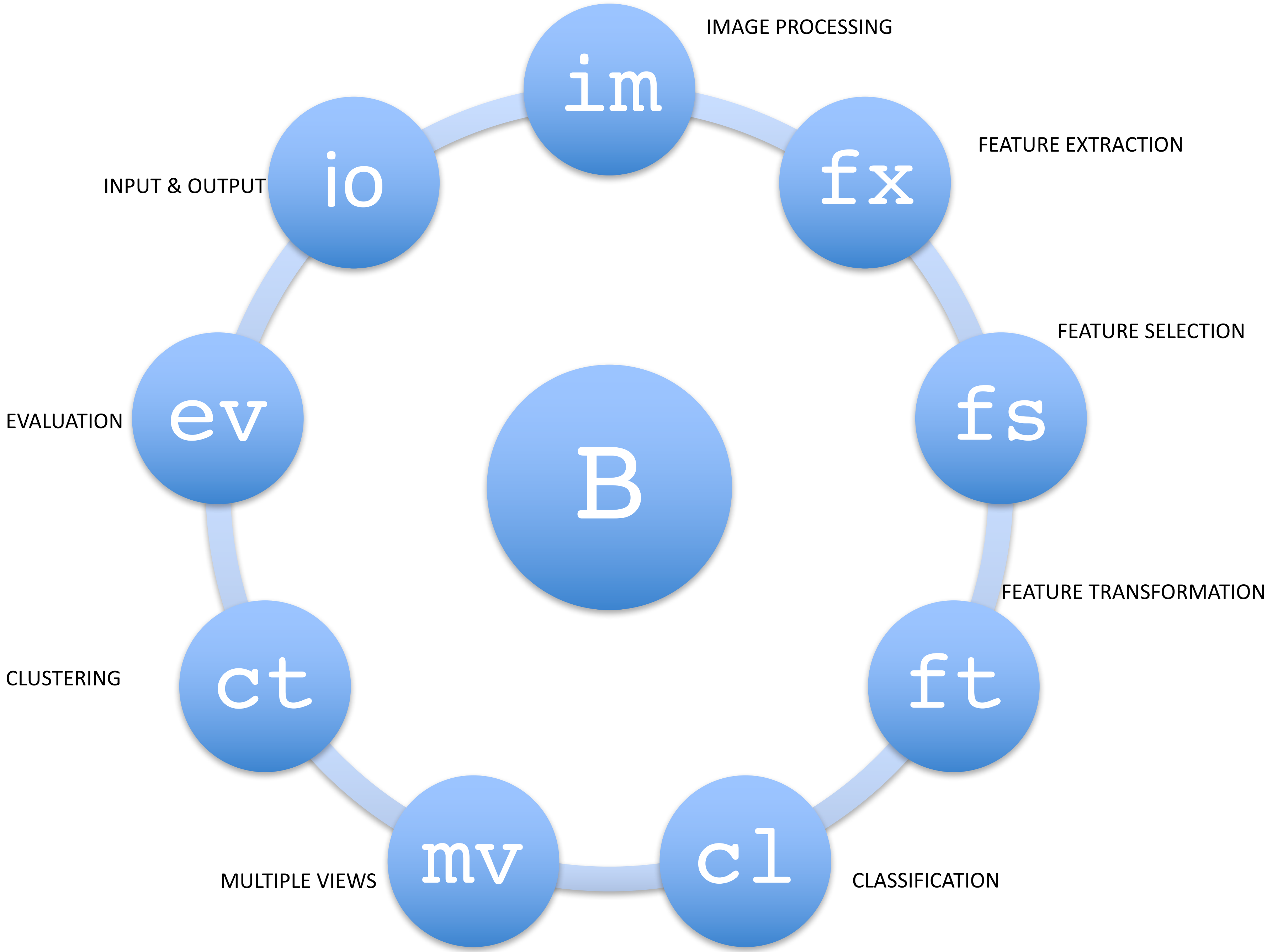
Balu: *Kid, I only got so much room up in this noggin... and it's fillin' up fast!*

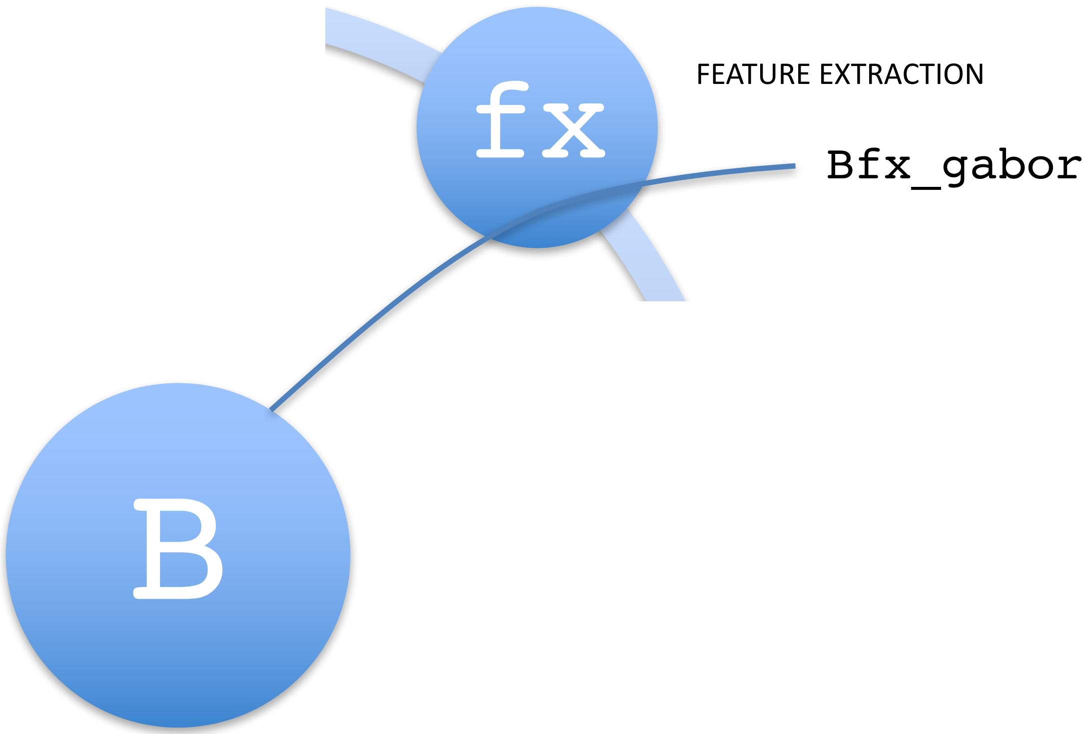
Mowgli: *You just don't understand.*

Balu: *All right. How's about layin' it out for me.*

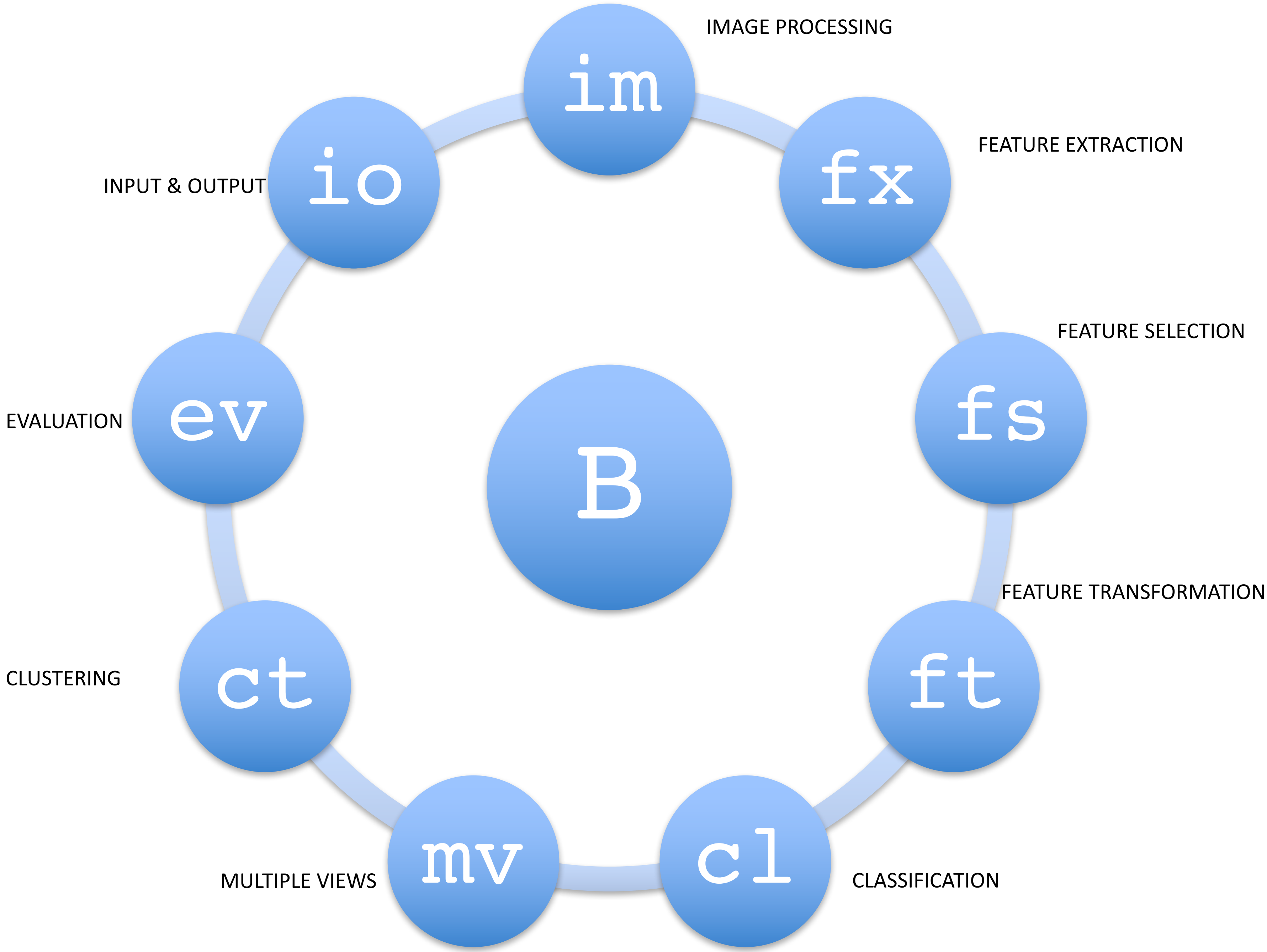
Balu could understand if Mowgli would explain and tell everything he knows... In this way, if we could extract a lot of features of an object we could recognize what the object is. This is the idea of this Toolbox!

Balu is the bear of The Jungle Book. He is the best friend of Mowgli. The story was written by Rudyard Kipling in 1894. Later in 1967, Disney produced a wonderful film

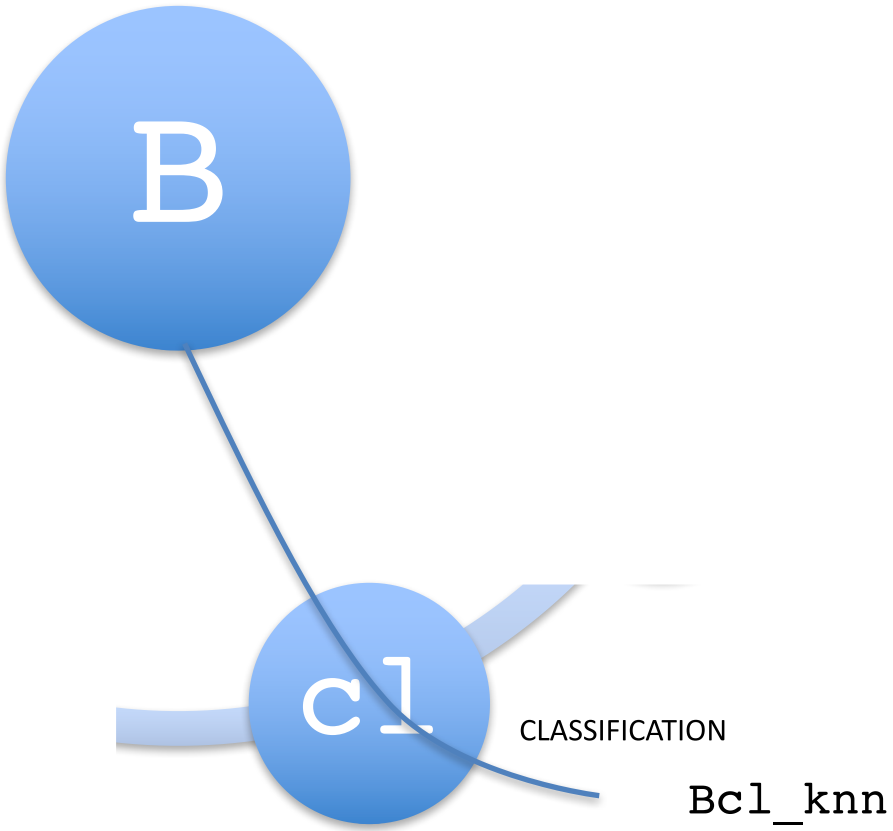

B
im
IMAGE PROCESSING
fx
FEATURE EXTRACTION
fs
FEATURE SELECTION
ft
FEATURE TRANSFORMATION
cl
CLASSIFICATION
mv
MULTIPLE VIEWS
ct
CLUSTERING
ev
EVALUATION
io
INPUT & OUTPUT



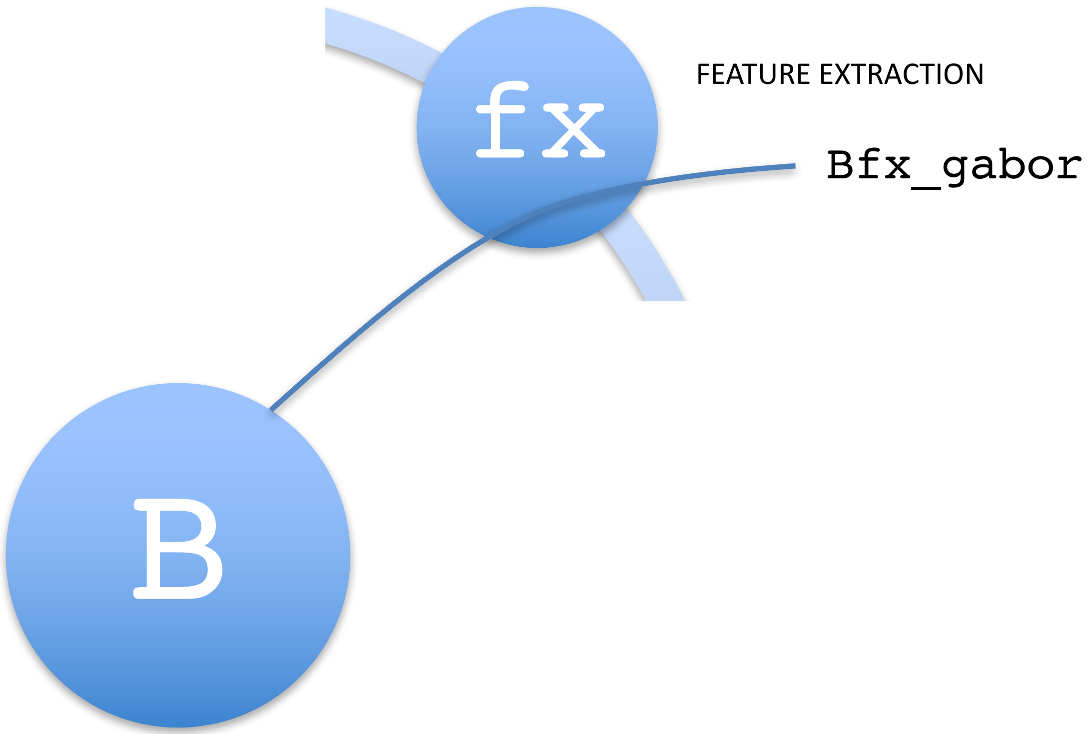
fx
FEATURE EXTRACTION
Bfx_gabor
B

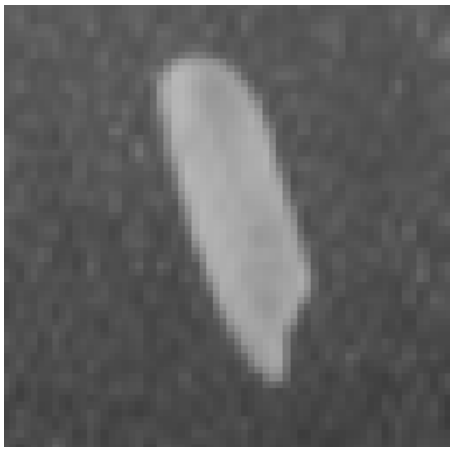


B
im
IMAGE PROCESSING
fx
FEATURE EXTRACTION
fs
FEATURE SELECTION
ft
FEATURE TRANSFORMATION
cl
CLASSIFICATION
mv
MULTIPLE VIEWS
ct
CLUSTERING
ev
EVALUATION
io
INPUT & OUTPUT

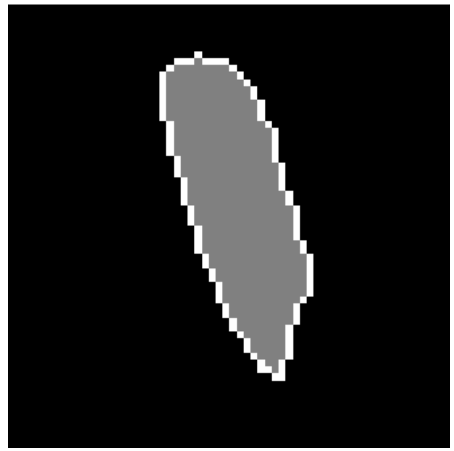

B
cl
CLASSIFICATION
Bcl_knn

Feature Extraction

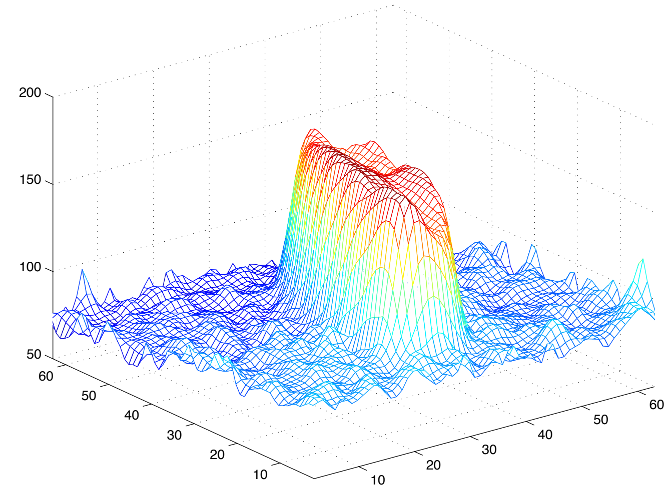

fx
FEATURE EXTRACTION
Bfx_gabor
B



a) Grayscale image



b) Segmentation



c) 3D representation of a)

There are two categories of features: Geometric Features and Intensity Features

Geometric Features give information about location, orientation, shape and size.

Intensity Features give information about how are the grayvalues.

How to extract Geometric Features with Balu

Command:

```
[X,Xn] = Bfx_name(R,options);
```

```
X:          row vector of extracted features
Xn:         name of each extracted feature

name:       name of the group of features
R:          binary image
options:    options of features (if any)
```

Example: BASIC GEOMETRIC FEATURES

```
I = imread('onerice.bmp');
R = I>120;

[X,Xn] = Bfx_basicgeo(R);
Bio_printfeatures(X,Xn);
```

```
 1 center of grav i [px]      29.299832
 2 center of grav j [px]      33.705193
 3 Height [px]                47.000000
 4 Width [px]                 22.000000
 5 Area [px]                  601.375000
 6 Perimeter [px]             107.500000
 7 Roundness                  0.653941
 8 Danielsson factor          2.253736
 9 Euler Number               1.000000
10 Equivalent Diameter [px]   27.570347
11 MajorAxisLength [px]       48.089929
12 MinorAxisLength [px]       16.112265
13 Orientation [grad]         -73.896848
14 Solidity                   0.937206
15 Extent                     0.577369
16 Eccentricity               0.942202
17 Convex Area [px]           637.000000
18 Filled Area [px]           597.000000
```

Example: HU MOMENTS

```
I = imread('onerice.bmp');
R = I>120;

[X,Xn] = Bfx_hugeo(R);
Bio_printfeatures(X,Xn);
```

```
1 Hu-moment 1                 0.269010
2 Hu-moment 2                 0.046196
3 Hu-moment 3                 0.000291
4 Hu-moment 4                 0.000081
5 Hu-moment 5                 0.000000
6 Hu-moment 6                 0.000013
7 Hu-moment 7                 0.000000
```

Example: ELLIPSE

```
I = imread('onerice.bmp');
R = I>120;

[X,Xn] = Bfx_fitellipse(R);
Bio_printfeatures(X,Xn);
```

```
1 Ellipse-centre i [px]     33.691981
2 Ellipse-centre j [px]     28.915583
3 Ellipse-minor ax [px]     7.380073
4 Ellipse-major ax [px]     24.353387
5 Ellipse-orient [rad]      -0.295048
6 Ellipse-eccentricity      0.303041
7 Ellipse-area [px]         564.637741
```

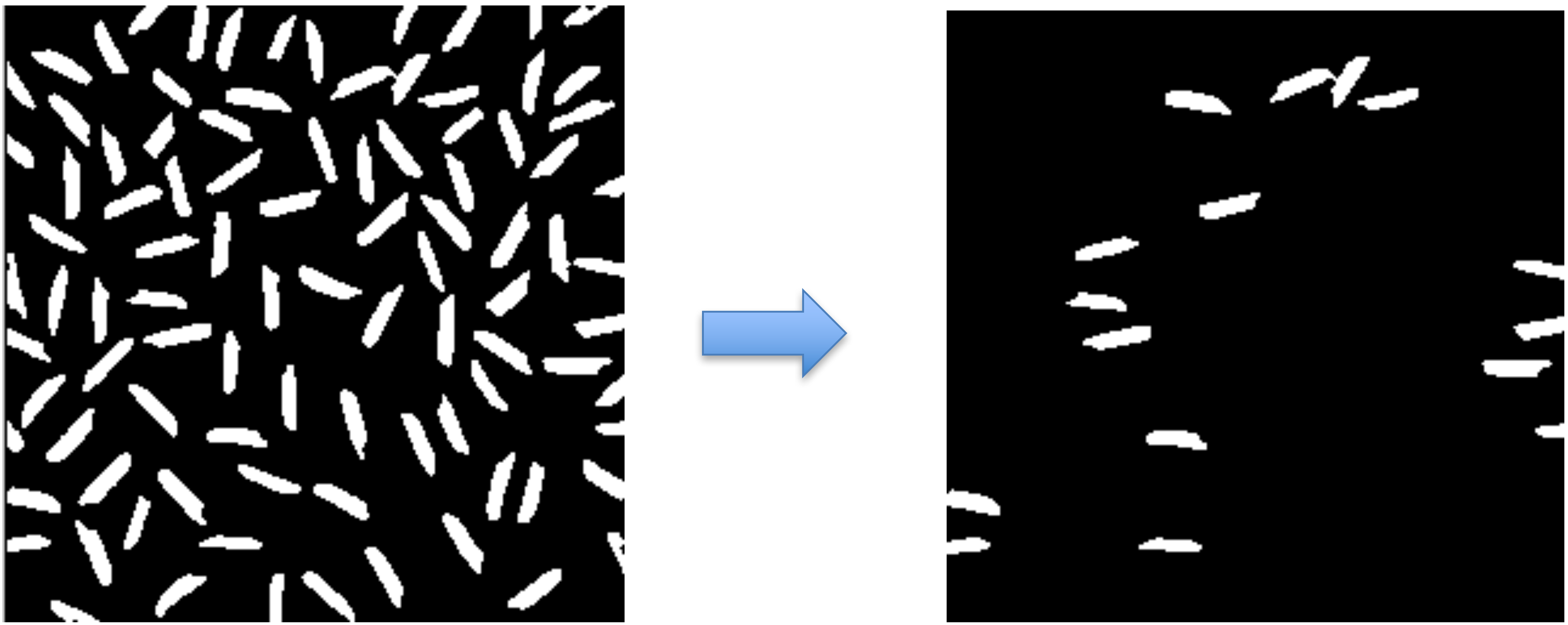
Example: SEVERAL GEOMETRIC FEATURES

```
% Input Image
I = imread('rice.png');
% Segmentation
[R,m] = Bim_segmowgli(I,ones(size(I)),40,1.5);


% Definition of features to be extracted
b(1).name = 'hugeo';       b(1).options.show=1;   % Hu moments
b(2).name = 'basicgeo';    b(2).options.show=1;   % basic geometric fetaures
options.b = b;

% Feature extraction
[X,Xn] = Bfx_geo(R,options);

% Processing after feature extraction
figure; hist(X(:,12));xlabel([Xn(12,:)])          % area histogram
ii = find(abs(X(:,20))<15);                        % rice orientation
K = zeros(size(R));                                % between -15 and 15 grad
for i=1:length(ii);K=or(K,R==ii(i));end
figure; imshow(K);title('abs(orientation)<15 grad')
```



Example: HOW TO WRITE A NEW BALU FUNCTION (for feature extraction)

```
% [X,Xn] = Bfx_centroid(R,options)
%
% Toolbox: Balu
%
%    Centroid of a region.
%
%    options.show    = 1 display mesagges.
%
%      X(1) is centroid-i, X(2) is centroid-j
%      Xn is the list of the n feature names.
%
%    Example (Centroid of a region)
%      I = imread('testimg1.jpg');     % input image
%      R = Bim_segbalu(I);             % segmentation
%      imshow(R);
%      options.show = 1;
%      [X,Xn] = Bfx_centroid(R,options);
%      Bio_printfeatures(X,Xn)
%

function [X,Xn] = Bfx_centroid(R,options)

if options.show == 1
    disp('--- extracting centroid...');
end
[Ireg,Jreg] = find(R==1);           % pixels in the region
ic = mean(Ireg);
jc = mean(Jreg);
X  = [ic jc];
Xn = [ 'Centroid i              '
       'Centroid j              ']; % 24 characters per name
if options.show == 1
    clf
    imshow(R)
    hold on
    plot(X(2),X(1),'rx')
    enterpause
end
```

This example show how to compute the centers of mass

How to extract Intensity Features with Balu

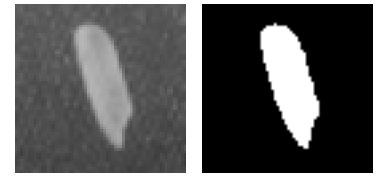
Command:

```
[X,Xn] = Bfx_name(I,R,options);
```

```
X:          row vector of extracted features
Xn:         name of each extracted feature

name:       name of the group of features
I:          grayscale image
R:          binary image -> [ ] for the whole image
options:    options of features (if any)
```

Example: BASIC INTENSITY FEATURES



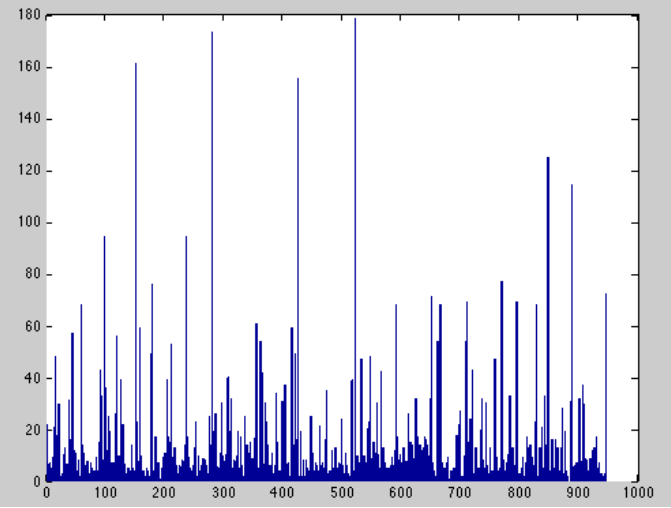
```
I = imread('onerice.bmp');
R = I>120;

options.mask = 5; % Gauss mask
options.show = 1; % Display results
[X,Xn] = Bfx_basicint(I,R,options);
Bio_printfeatures(X,Xn);
```

```
1 Intensity Mean              161.132328
2 Intensity StdDev            13.082633
3 Intensity Kurtosis          4.044382
4 Intensity Skewness          -1.096439
5 Mean Laplacian              -6.871022
6 Mean Boundary Gradient      41.875637
```

Example: LBP

```
I = imread('face0304.bmp');
I = imresize(I,[110 90]);
options.vdiv = 4;
options.hdiv = 4;
options.samples  = 8;
options.mappingtype = 'u2';
[X,Xn] = Bfx_lbp(I,[],options);
bar(X)
```



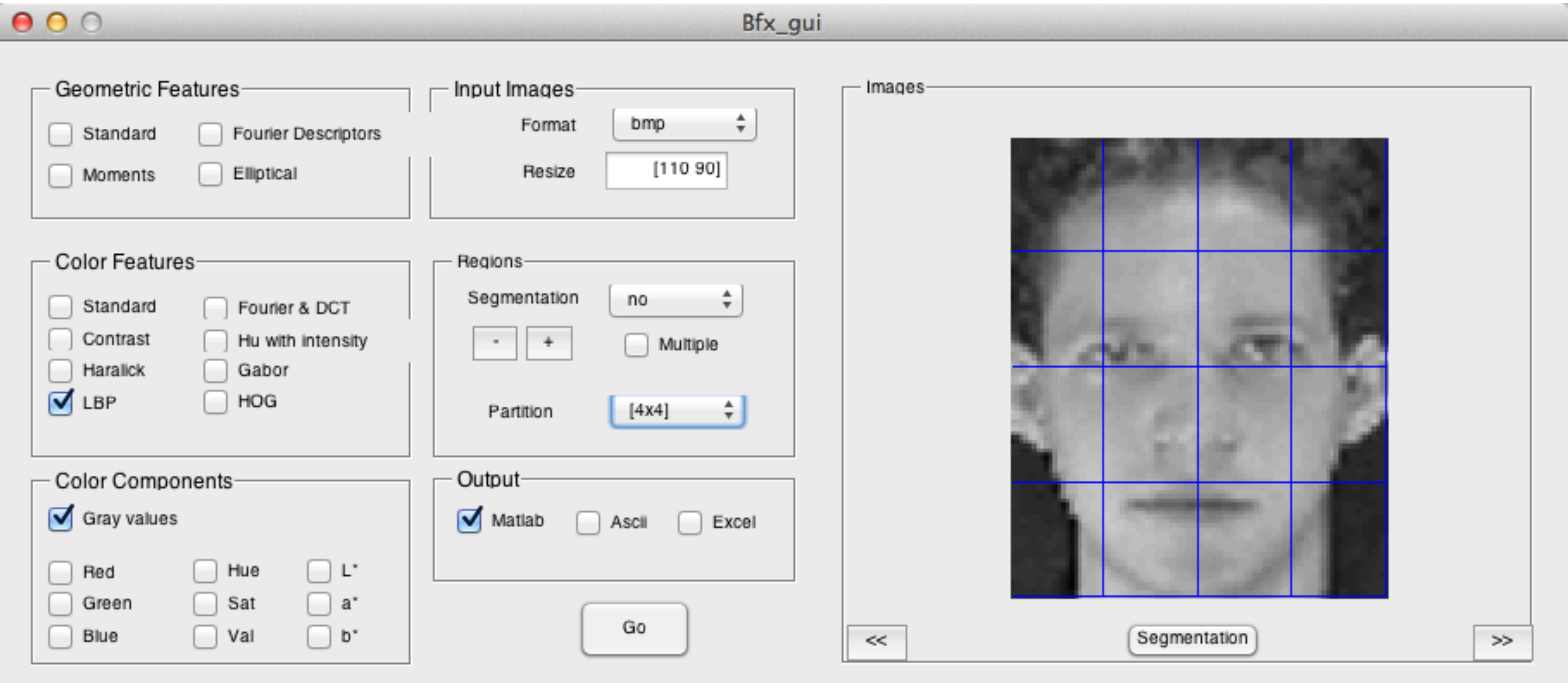
Example: SEVERAL INTENSITY FEATURES

See help Bfx_int

Graphic User Interface Bfx_gui

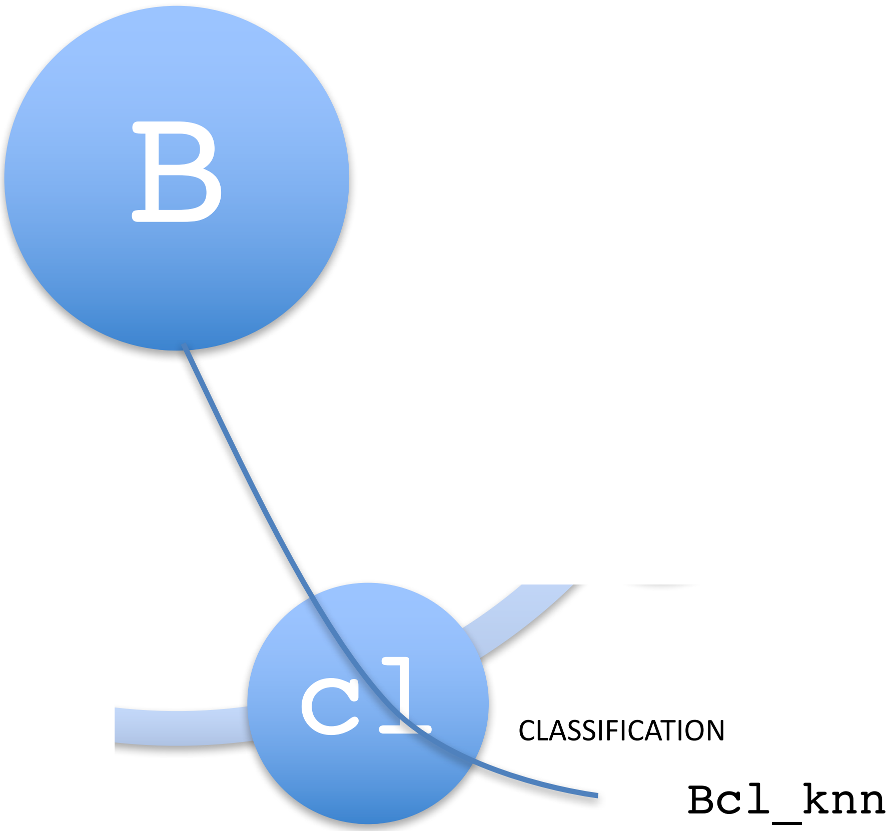
It is used to extract features in many image

1. go to the folder of the images
2. Bfx_gui <Enter>



Press go, drink a coffee and wait!
The results will be stored in file Bfx_results.mat

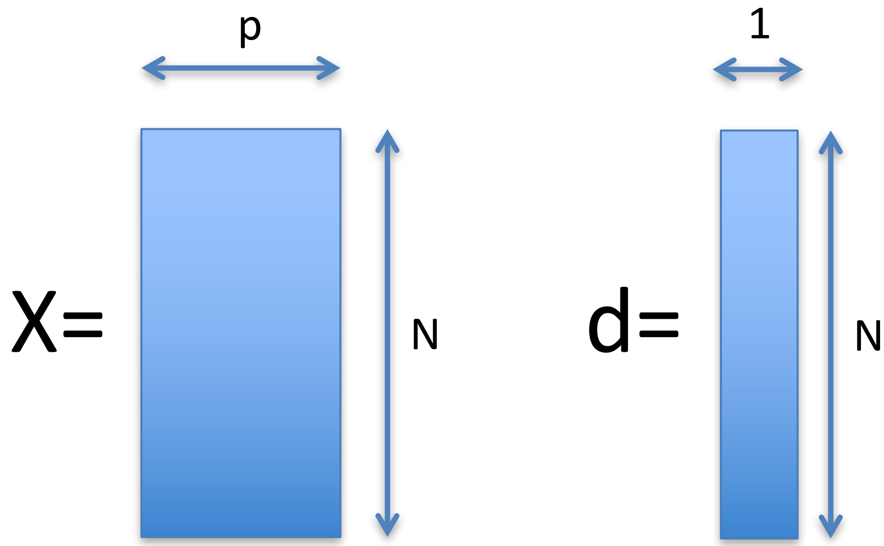
Classification


B
cl
CLASSIFICATION
Bcl_knn

How to use
the classifiers
of Balu

1.YOU NEED:

Training Data

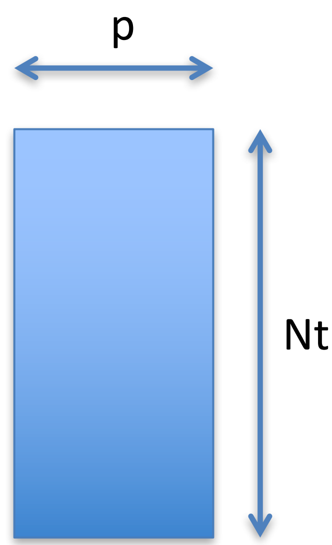


X: N samples, each sample is a vector of p features

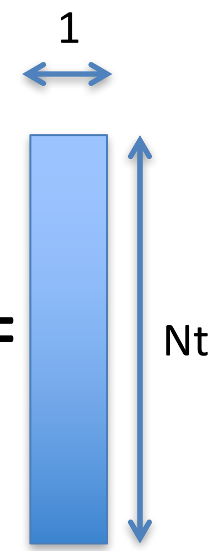
d: Labels, annotations (class for each sample)

1.YOU NEED:

Testing Data Xt=



dt=



Xt: Nt samples (of p features)

dt: Nt Labels, annotations

1.YOU NEED:

Classifier:

- Name of the classifier (KNN, LDA, SVM…)
- Options of the classifiers (k for KNN, kernel for SVM, etc…)

2. SYNTAX (Training & Testing)

ds = Bcl_*name*(X,d,Xt,options)

Example 1:

```
% KNN with k = 5
options.k = 5;
ds = Bcl_knn(X,d,Xt,options);
```

Example 2:

```
% LDA with p = (1/4 3/4);
options.p = [0.25 0.75];
ds = Bcl_lda(X,d,Xt,options);
```

ds is a Nt x 1 vector with the classification of each testing sample. To compute the performance:

```
p = Bev_performance(ds,dt);
```

3. SYNTAX (Training only)

op = Bcl_*name*(X,d,options)

All parameters of the classifier that were estimated

Example 1:

```
% KNN with k = 5
options.k = 5;
op = Bcl_knn(X,d,options);
```

Example 2:

```
% LDA with p = (1/4 3/4);
options.p = [0.25 0.75];
op = Bcl_lda(X,d, options);
```

4. SYNTAX (Testing only)

You need the parameters ‘op’ computed in last step.

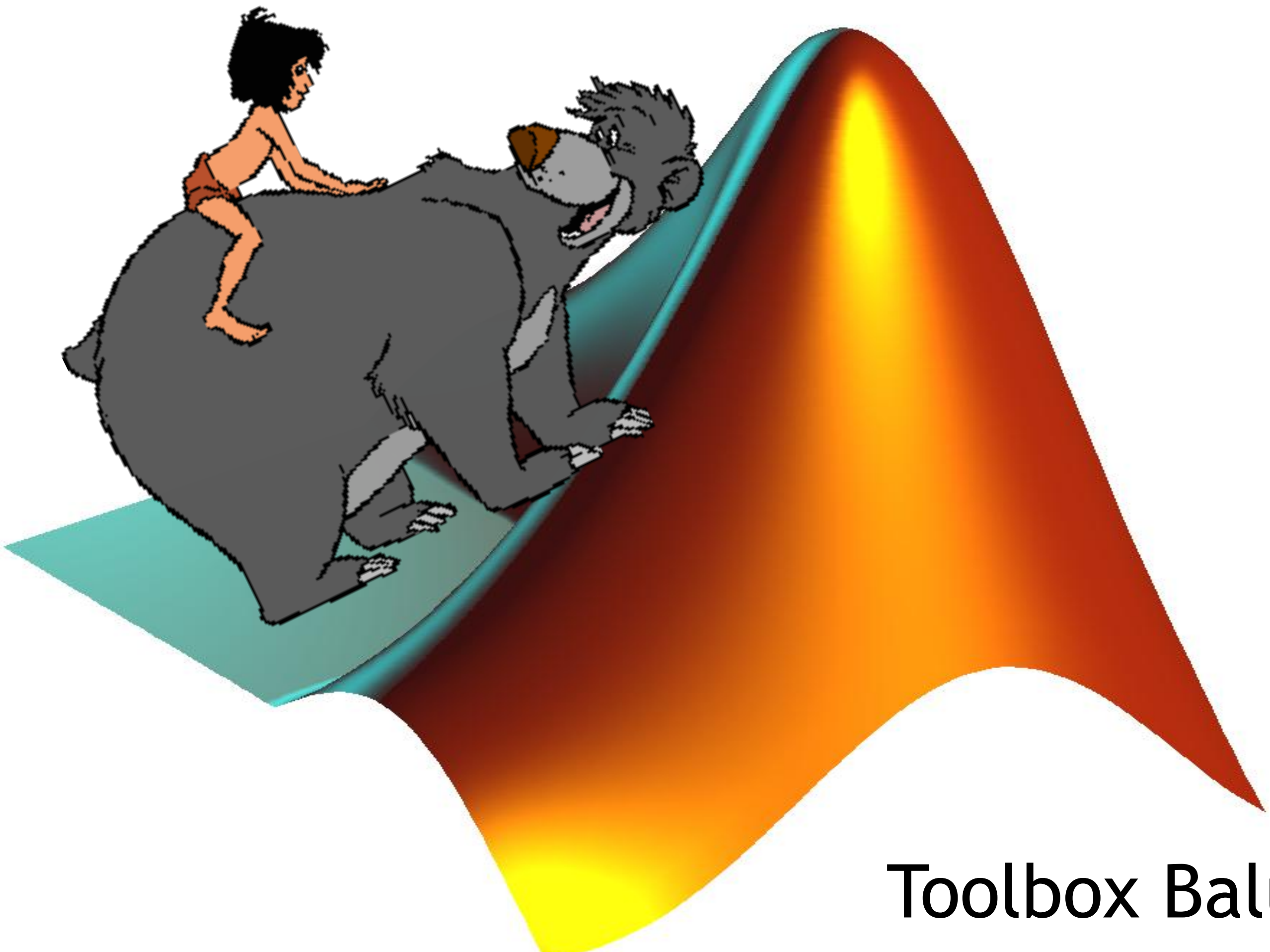
ds = Bcl_*name*(Xt,op)

Example 1:

```
% KNN with k = 5
ds = Bcl_knn(Xt,op);
```

Example 2:

```
% LDA with p = (1/4 3/4);
op = Bcl_lda(Xt,op);
```



Toolbox Balu

http://dmery.ing.puc.cl/index.php/balu/