Quickreference for Balu Toolbox Matlab

|  |  |
| --- | --- |
| **Graphic user interfaces**  Bfx\_gui  Bcl\_gui    **Image segmentation**  [R,E,J] = Bim\_segbalu(I,p)  [F,m] = Bim\_segmowgli(J,R,Amin,sig)  **Feature extraction**  [X,Xn] = Bfx\_basicgeo(R,options)  [X,Xn] = Bfx\_basicint(I,R,options)  [X,Xn] = Bfx\_hugeo(R,options)  [X,Xn] = Bfx\_flusser(R,options)  [X,Xn] = Bfx\_gupta(R,options)  [X,Xn] = Bfx\_fourierdes(R,options)  [X,Xn] = Bfx\_fitellipse(R,options)  [X,Xn] = Bfx\_clp(I,R,options)  [X,Xn] = Bfx\_haralick(I,R,options)  [X,Xn] = Bfx\_huint(I,R,options)  [X,Xn] = Bfx\_gabor(I,R,options)  [X,Xn] = Bfx\_lbp(I,R,options)  [X,Xn] = Bfx\_bsif(I,R,options)  **Input/Output**  Bio\_imshow(I,p)  Bio\_edgeview(I,E)  Bio\_plotfeatures(f,d,fn)  Bio\_printfeatures(fn,f,fu)  Bio\_loadimg(f,i)  **Data generation**  [X,d] = Bds\_gaussgen(m,s,n)  **Data selection**  [X1,d1,X2,d2] = Bnostratify(X,d,s)  [X1,d1,X2,d2] = Bstratify(X,d,s)  [Xb,db,Xnb,dnb] = Bbootstrapsample(X,d,N)  **Feature transformation**  [Xn,a,b] = Bfnorm(X,normtype)  [Y,lambda,A,Xs] = Bpca(X,m) | **Feature selection**  selec = Bfsfs(X,d,m,show,method,param1,param2,...)  selec = Bfsfosmod(X,m,show)  selec = Bfsrank(X,d,m,criterion)  selec = Bfslsef(X,m,show)  selec = Bfsclean(X,show)  selec = Bfsel(X,d,m,s,show)  selec = Bfsexsearch(X,d,m,show,method,param1,param2,...)  [T,U,P,Q,W,B] = Bplsr(X,d,m)  **Classifier Design**  ds = Bclass(X,d,Xt,method,varargin)  ds = Badaboost(X,d,Xt,T)  ds = Bbagging(X,d,Xt,B,method,param1,param2,...)  ds = Bbayes2(X,d,Xt,p,show) \*\*\* only two features & two classes  ds = Bboosting(X,d,Xt,s)  ds = Bdet21(X,d,Xt,method) \*\*\* only two features & two classes  ds = Bdet22(X,d,Xt,method) \*\*\* only two features & two classes  ds = Bdmin(X,d,Xt)  ds = Bknn(X,d,Xt,k)  ds = Blda(X,d,Xt)  ds = Bmaha(X,d,Xt)  ds = Bpnn(X,d,Xt)  ds = Bqda(X,d,Xt)  ds = Bsvm(X,d,Xt,kernelnum,kernelpar,cte)  ds = Bsvm2(X,d,Xt,kernel)  ds = Bviola(X,d,Xt,T,show)  ds = Bnnglm(X,d,Xt,method,iter)  ds = Bclassifier(X,d,Xt,Bsclass)  ds = Bensemble(X,d,Xt,bclass,tensemble,param,show)  [p,th,ys] = Bweakc(x,y,xt)  **Performance evaluation**  [T,p] = Bconfusion(d,ds,nn)  p = Bperformance(d,ds,nn)  [TT,per] = Bholdout(X,d,s,strat,method,param1,param2...)  [TT,per] = Bcrossval(X,d,v,method,param1,param2,...)  [TT,per] = Bjackknife(X,d,method,param1,param2,...)  [TT,per] = Bbootstrap(X,d,B,method,param1,param2,...)  [TT,per] = Bbootstrap632(X,d,B,method,param,param2,...)  [per,ci] = Bcrossvalidation(X,d,v,Bsclass) |

© GRIMA, DCC-UC, http://dmery.ing.puc.cl