BUSAT: A MATLAB Toolbox for Breast Ultrasound Image Analysis

Arturo Rodríguez-Cristerna[1](✉), Wilfrido Gómez-Flores[1],
and Wagner Coelho de Albuquerque-Pereira[2]

[1] Center for Research and Advanced Studies of the National Polytechnic Institute,
Ciudad Victoria, Tamaulipas, Mexico
{arodriguez,wgomez}@tamps.cinvestav.mx

[2] Biomedical Engineering Program, COPPE, Federal University of Rio de Janeiro,
Rio de Janeiro, Brazil
wagner.coelho@ufrj.br

Abstract. This paper presents the Breast Ultrasound Analysis Toolbox
(BUSAT) for MATLAB, which contains 62 functions to perform image
preprocessing, lesion segmentation, feature extraction, and lesion classifi-
cation. BUSAT is useful to codify programs for computer-aided diagnosis
(CAD) purposes in reduced time; hence, to replicate several approaches
proposed in literature is feasible. We provide the implementation of a
CAD system to classify breast lesions into benign and malignant classes
and an example to evaluate the classification performance. BUSAT could
be downloaded from the following permanent link: http://www.tamps.
cinvestav.mx/~wgomez/downloads.html.

Keywords: Breast ultrasound · Image analysis · Computer-aided diag-
nosis · MATLAB

1 Introduction

Breast cancer is the most frequently diagnosed cancer and the leading cause of
cancer death among women worldwide [1]. Hence, early diagnosis is a crucial
factor in breast cancer treatment, where medical images are important sources
of diagnostic information. Currently, breast ultrasound (BUS) is an important
coadjuvant technique to mammography (x-ray) in patients with palpable masses
and normal or inconclusive mammogram findings [2]. Also, BUS images are
particularly effective in distinguishing cystic from solid lesions and are useful for
differentiating between benign and malignant tumors [3].

In order to assist radiologists in the BUS image interpretation, computer-
aided diagnosis (CAD) systems have emerged as a 'second reader' for analyzing
the images by using computational approaches. Generally, the pipeline of a CAD
system involves four basic stages: image preprocessing, lesion segmentation, fea-
ture extraction, and lesion classification [4]. Then, radiologists can take the CAD
outcome as a second opinion and make a more conclusive diagnosis for reducing
unnecessary biopsies in benign cases [5].

Image preprocessing commonly increases the contrast between the lesion region and its background, and also considers low-pass filtering to reduce the speckle artifact. Next, BUS segmentation procedure separates the lesion region from its background and other tissue structures. Thereafter, from segmented lesions, morphological and texture features are usually computed and to improve the between-class discrimination, relevant features are selected. These features represent the classifier inputs for distinguishing the lesions into benign and malignant classes [4].

In literature, a plethora of approaches have been proposed to address each stage of CAD systems for BUS images. In this sense, Cheng et al. [4] and Huang et al. [6] presented comprehensive surveys related to BUS image analysis. Despite the large quantity of proposed approaches, to get useful computational implementations for research purposes is usually difficult, because the source codes or programs are not commonly shared by the authors.

Hence, we introduce a MATLAB (The MathWorks, Natick, Massachusetts, USA) [7] toolbox for BUS image analysis, aiming to share with the research community the efforts that we made to implement several methods to develop CAD systems for breast ultrasound. The toolbox is composed of 62 functions divided into four sections: image preprocessing, lesion segmentation, feature extraction, and classification. This toolbox could be downloaded from our permanent link http://www.tamps.cinvestav.mx/~wgomez/downloads.html.

2   Toolbox Organization

The Breast Ultrasound Analysis Toolbox (BUSAT) has 62 functions oriented to image preprocessing (contrast enhancement, despeckling, and domain transformation), lesion segmentation (semi-automatic and fully-automatic methods), feature extraction (morphological, texture, and BI-RADS lexicon), and classification (linear and non-linear classifiers). Figure 1 illustrates the general organization of BUSAT and the list of available functions.

It is worth mentioning that all the functions were codified by our research group based on several articles from literature; hence, all the implemented methods have theoretical basis. In addition, several functions take advantage of some methods developed by other research groups to guarantee the quality of the results, for instance, LIBSVM to train Support Vector Machines [8], minimum redundancy maximum relevance (mRMR) for feature selection [9], etc.

On the other hand, the main BUSAT directory contains the following six subfolders:

– **Data:** contains data files and test images to run the examples of the toolbox.
– **Preprocessing:** 13 functions for contrast enhancement, speckle filtering, and domain transformation.
– **Segmentation:** four functions for lesion segmentation.
– **Features:** 29 functions for computing morphological, texture, and BI-RADS features.
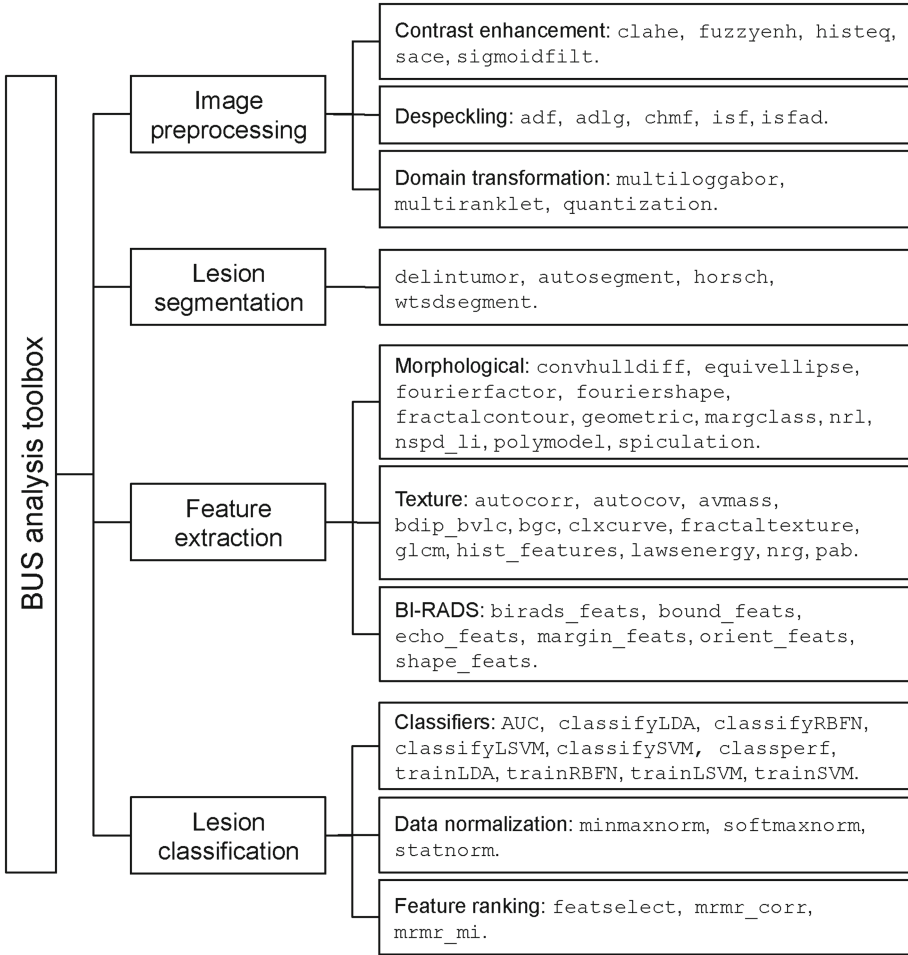
Fig. 1. Organization of the BUS analysis toolbox and list of functions.

– **Classification:** 16 functions for lesion classification in benign and malignant classes.
– **C functions:** 21 compiled C code functions that are used by several functions of the toolbox.

3   Toolbox Usage

3.1   Installation

To start using BUSAT, the script RUN_ME_FIRST should be firstly run to add all the toolbox directories to the MATLAB search path.

3.2 Help Topics

To display the organization BUSAT, type in the MATLAB Command Window the statement `help Contents`. Note that every listed function has a hyperlink to its own help topics. Also, the user can consult the help topics of a specific function by typing the statement `help` followed by the name of the function as illustrated in Fig. 2. Observe that help topics are displayed in three parts: the syntax explanation of the function, an illustrative example, and the reference or bibliography for theoretical details. Also, hyperlinks to similar functions are showed.

```
>> help isfad
  isfad Interference-based speckle filter followed by anisotropic diffusion.
    J = isfad(I,W,MAXITER) performs the interference-based speckle filter
    followed by anisotropic diffusion for despeckling a breast ultrasound
    image I, where W is the window size of the median filter, and
    MAXITER is the maximum number of iterations for diffusion.

    Example:
    -------
    I = imread('BUSreal.tif');
    J = isfad(I,10,500);
    figure;
    subplot 121; imshow(I); title('Original Image');
    subplot 122; imshow(mat2gray(J)); title('Filtered Image');

    See also adf adlg chmf isf


    Reference:
    ---------
    F. M. Cardoso, M. M. S. Matsumoto, S. S. Furuie "Edge-preserving speckle
    texture removal by interference-based speckle filtering followed by
    anisotropic diffusion," Ultrasound in Medicine and Biology, vol. 38,
    no. 8, pp. 1414-1428, 2012.
```

Fig. 2. Example of help topics for a specific function.

3.3 Running Examples

Every function in BUSAT could be tested by running the example provided in the help topics. This could be performed by copying and pasting the example text on the Command Window. In the case of *image preprocessing* and *lesion segmentation* functions, both the original and the processed images are showed. For instance, images showed in Fig. 3 are displayed after running the example code in Fig. 2.

3.4 Special Considerations

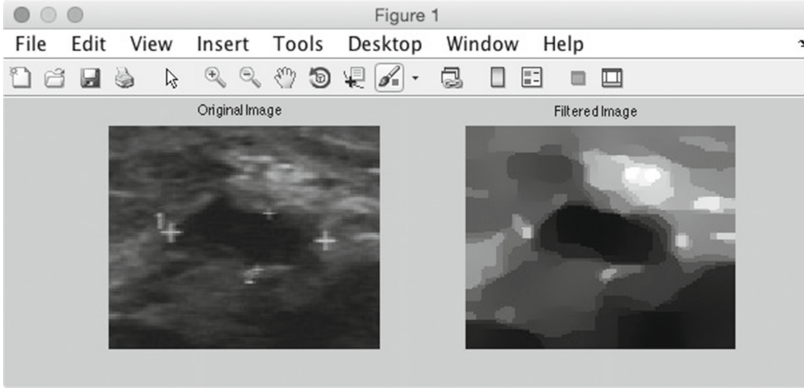Two special considerations should be taken into account:

Fig. 3. Example of a BUS image despeckled by `isfad` function.

1. **C code functions:** despite BUSAT provides compiled C code functions
   (called *mex functions*) for Linux, Mac OS and Windows using 64-bits proces-
   sors, in some operative systems they should be recompiled from the source
   codes by using the MATLAB `mex` function. These source codes are provided
   within the directory `Source_C_codes`.
2. **Parallel Computing Toolbox:** to speed-up the execution of the functions
   `autosegment`, `trainLSVM`, `trainSVM`, `trainRBF`, and `featselect`, the
   parallel pool is automatically open if the MATLAB *Parallel Computing Tool-
   box* is available, otherwise, the functions are sequentially executed.

4   Practical Examples

4.1   Building a CAD System

BUSAT is useful to quickly build a CAD system by following the pipeline in
Fig. 4. Note that distinct functions of contrast enhancement, speckle filtering,
lesion segmentation, feature extraction, and lesion classification could be com-
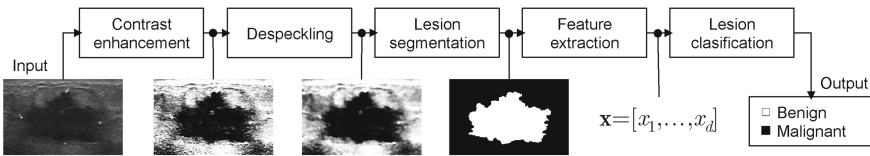bined to create a specific CAD system.

Fig. 4. Conventional pipeline of a CAD system for BUS images.

Herein, BUSAT is used to exemplify the implementation of a CAD system
that uses five morphological features and linear classification [10]. The imple-
mented CAD system uses the `wtsdsegment` function to segment the breast

lesion. This function already considers the image preprocessing, where contrast enhancement is performed by `sace` function, whereas speckle filtering is performed by `chmf` function. Thereafter, the segmentation algorithm based on watershed transformation is applied to get the lesion contour [11]. Next, five morphological features are computed: elliptic-normalized skeleton, lesion orientation, number of substantial protuberances and depressions, depth-to-width ratio, and overlap ratio. Finally, `classifyLDA` function classifies the lesion in benign and malignant classes by using linear discriminant analysis (LDA). Obviously, the LDA classifier should be previously trained with the `trainLDA` function to create the prediction model. Then, the MATLAB program that implements the CAD system is written as follows:

```
% Read breast ultrasound image
I = imread('bus_image.tif');
% Load training data: normalization data and LDA model
load('cad.mat');
% Image preprocessing and lesion segmentation
BW = wtsdsegment(I);
% Compute five morphological features:
% 1. Elliptic normalized-skeleton
x1 = equivellipse(BW,'ens');
% 2. Lesion orientation
x2 = equivellipse(BW,'angle');
% 3. Number of substantial protuberances-depressions
x3 = nspd_li(BW,'nspd');
% 4. Depth-to-width ratio
x4 = geometric(BW,'dwr');
% 5. Overlap ratio
x5 = convhulldiff(BW,'or');
% Complete feature vector
x = [x1 x2 x3 x4 x5];
% Normalization: mean (mn) and standard deviation (sd)
xn = softmaxnorm(x,[mn;sd]);
% Lesion classification with LDA model
C = classifyLDA(xn,lda_model);
```

4.2   Evaluating a CAD System

When a CAD system is developed, it is necessary to evaluate its classification performance in terms of some indices such as accuracy, sensitivity, specificity, area under the ROC curve, etc.

Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a feature space with n observations, where the ith observation is a d-dimensional feature vector denoted by $\mathbf{x}_i = [x_{i,1}, \ldots, x_{i,d}]$. Also, the observation $\mathbf{x}_i$ is associated to a class label $y_i \in \{1, 2\}$, where 1 and 2 denote benign and malignant lesions, respectively. Note that this kind of labeling

is required by the training functions, although depending on the classifier the labels are adjusted. For instance, for the SVM classifier, the label $y = 1$ becomes $y = -1$ and the label $y = 2$ becomes $y = +1$.

Then, to perform CAD assessment, from the $\mathcal{X}$ set, training and test sets should be created, where the former is used to generate the prediction model and the latter is used to evaluate the classifier generalization. In addition, if the classifier requires hyperparameters, a grid-search scheme and k-fold cross validation method are automatically performed by the training functions to tune such parameters. For instance, the function `trainSVM` adjusts both the soft margin parameter C and the Gaussian kernel parameter γ, if they are not introduced in the input arguments of the function.

BUSAT contains the `classperf` function to evaluate the classification performance of a CAD system. Suppose that a user generates a feature matrix **X** of size $n \times d$ and a target vector **Y** of size $n \times 1$. Also, suppose that the CAD's classifier is based on SVM with Gaussian kernel. Then, the following MATLAB program implements the evaluation of a CAD system:

```
% Split feature space into training and test sets
ho = crossvalind('HoldOut',Y,0.2);  % 20% for test set
% Normalize training set
[Xtr,mn,sd] = softmaxnorm(X(ho,:));
% Normalize test set
Xtt = softmaxnorm(X(not(ho),:),[mn;sd]);
% Get training targets
Ytr = Y(ho,:);
% Get test targets
Ytt = Y(not(ho),:);
% Train SVM prediction model
svm_model = trainSVM(Xtr,Ytr);
% Classify test set with the SVM prediction model
C = classifySVM(Xtt,svm_model);
% Evaluate classification performance
p = classperf(C.Labels,Ytt);
```

5   Experimental Results

BUSAT contains three classifiers for distinguishing between benign and malignant lesions: linear discriminant analysis (LDA), support vector machine (SVM) with Gaussian kernel, and radial basis function network (RBFN). These classifiers are evaluated within a CAD system to determine which method performs better in terms of the indices Matthews correlation coefficient (MCC), area under the ROC curve (AUC), accuracy (ACC), sensitivity (SEN), and specificity (SPE) [12].

The BUS dataset considered 1,128 cases from 659 female patients acquired during routine breast diagnostic procedures at the National Cancer Institute

(INCa) of Rio de Janeiro, Brazil. All the cases were histopathologically proven by biopsy, where 781 images presented benign lesions and 347 images had malignant tumors. The images were collected from three ultrasound scanners with linear transducer arrays with frequencies between 7.5 and 12 MHz: Logiq 7 (GE Medical System Inc.), Logiq 5 (GE Medical System Inc.), and Sonoline Sienna (Siemens).

The entire dataset was segmented by the `wtsdsegment` function. Next, 25 morphological and texture features were computed, which are summarized in Table 1. The feature space was randomly split in training (90%) and test (10%) sets, which were normalized by the `softmaxnorm` function. Thereafter, LDA, SVM, and RBFN classifiers were trained by the functions `trainLDA`, `trainSVM`, and `trainRBFN`, respectively. It is worth mentioning that `trainSVM` and `trainRBFN` functions perform grid-search and k-fold cross validation method (with $k = 10$) to tune their parameters. In the case of the SVM, the C and γ parameters are adjusted, whereas for the RBFN, the number of hidden units is determined. Finally, the test set was classified by the functions `classifyLDA`, `classifySVM`, and `classifyRBFN`, and the classification performance of each classifier was evaluated by the `classperf` function. For statistical analysis, 50 independent runs of training-testing procedure was performed.

Table 1. Computed features for lesion classification. $\mathcal{M}$ and $\mathcal{T}$ denote morphological and texture features, respectively. Symbol # denotes number of features.

Type	Technique	Function	#
$\mathcal{M}$	Equivalent ellipse	`equivellipse`	5
$\mathcal{M}$	Fractal analysis	`fractalcontour`	2
$\mathcal{M}$	Geometry	`geometric`	1
$\mathcal{M}$	Anfractuosity	`margclass`	1
$\mathcal{M}$	Signature	`nrl`	1
$\mathcal{T}$	Autocorrelation	`autocorr`	1
$\mathcal{T}$	Gray-level average	`avmass`	2
$\mathcal{T}$	Gray-level co-occurrence matrix	`glcm`	9
$\mathcal{T}$	Laws' energy measures	`lawsenergy`	2
$\mathcal{T}$	Posterior acoustic behavior	`pab`	1

Table 2 summarizes the classification performance results obtained by the three evaluated classifiers. Besides, Table 3 shows the one-way analysis of variance (ANOVA) results to test whether the mean values between compared classifiers are different at $\alpha = 0.05$. Also, the Scheffe's method determines if there is statistical significance between two classifiers.

It is notable that the three classifiers did not present statistical differences in terms of MCC and AUC indices, that is, they are capable of distinguishing adequately between benign and malignant cases. However, the SVM classifier

Table 2. Classification performance results (mean ± standard deviation).

Classifier	MCC	AUC	ACC	SEN	SPE
LDA	0.75 ± 0.05	0.89 ± 0.02	0.88 ± 0.02	0.85 ± 0.05	0.90 ± 0.03
SVM	0.76 ± 0.05	0.89 ± 0.02	0.89 ± 0.02	0.90 ± 0.04	0.89 ± 0.03
RBFN	0.76 ± 0.06	0.90 ± 0.02	0.86 ± 0.03	0.79 ± 0.06	0.94 ± 0.02

Table 3. p-values of the statistical comparison between classifiers. Symbol $(-)$ denotes that groups are not statistically significant different (i.e., $p > 0.05$), contrarily symbol $(+)$ indicates that groups are statistically significant different (i.e., $p < 0.05$).

Comparison	MCC	AUC	ACC	SEN	SPE
LDA vs. SVM	$0.28^{(-)}$	$0.67^{(-)}$	$0.03^{(+)}$	$0.00^{(+)}$	$0.09^{(-)}$
LDA vs. RBFN	$0.70^{(-)}$	$0.20^{(-)}$	$0.09^{(-)}$	$0.00^{(+)}$	$0.00^{(+)}$
SVM vs. RBFN	$0.75^{(-)}$	$0.67^{(-)}$	$0.00^{(+)}$	$0.00^{(+)}$	$0.00^{(+)}$

outperformed its counterparts in terms of sensitivity (SEN = 0.90) and accuracy (ACC = 0.89), whereas the RBFN classifier obtained the best results in terms of specificity (SPE = 0.94). These results pointed out that the SVM classifier is adequate to be implemented within a CAD system for BUS images.

6 Conclusions

This paper presented the Breast Ultrasound Analysis Toolbox (BUSAT) for MATLAB, which contains several approaches proposed in literature to perform image preprocessing (contrast enhancement and speckle filtering), lesion segmentation (semi-automatic and fully-automatic methods), feature extraction (morphological, texture, and BI-RADS lexicon), and classification (linear and non-linear classifiers).

We presented the experimental results of the evaluation of three classifiers (LDA, SVM, and RBFN) to distinguish between benign and malignant cases, where SVM presented an adequate classification performance. Obviously, the configuration of the CAD system could lead to different classification results, that is, the image preprocessing techniques, the segmentation method, and the computed features impact on the lesion classification. Thus, the potential of BUSAT is the versatility to build and evaluate different configurations of CAD systems in reduced time.

To the best of our knowledge, BUSAT is the first toolbox intended to provide to the research community an easy and quick way to codify programs for computer-aided diagnosis for breast ultrasound. In addition, because the source codes are available to the users, it is possible to modify the functions in order to enhance the implemented methods or reuse code in new functions. Feature work considers to increase the number of implemented methods, for instance, new multiclass classifiers for BI-RADS categorization.

References

1. Ferlay, J., Soerjomataram, I., Dikshit, R., Eser, S., Mathers, C., Rebelo, M., Parkin, D., Forman, D., Bray, F.: Cancer incidence and mortality worldwide: sources, methods and major patterns in globocan 2012. Int. J. Cancer **136**(5), E359–E386 (2015)
2. Kelly, K.M., Dean, J., Comulada, W.S., Lee, S.J.: Breast cancer detection using automated whole breast ultrasound and mammography in radiographically dense breasts. Eur. Radiol. **20**(3), 734–742 (2010)
3. Stavros, A.T., Thickman, D., Rapp, C.L., Dennis, M.A., Parker, S.H., Sisney, G.A.: Solid breast nodules: use of sonography to distinguish between benign and malignant lesions. Radiology **196**(1), 123–134 (1995)
4. Cheng, H.D., Shan, J., Ju, W., Guo, Y., Zhang, L.: Automated breast cancer detection and classification using ultrasound images: a survey. Pattern Recogn. **43**, 299–317 (2010)
5. Drukker, K., Gruszauskas, N.P., Sennett, C.A., Giger, M.L.: Breast us computer-aided diagnosis workstation: performance with a large clinical diagnostic population. Radiology **248**(2), 392–397 (2008)
6. Huang, Q., Luo, Y., Zhang, Q.: Breast ultrasound image segmentation: a survey. Int. J. Comput. Assist. Radiol. Surg. **12**(3), 493–507 (2017)
7. MathWorks: Matlab. the language of technical computing
8. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) **2**(3), 27 (2011)
9. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. **27**(8), 1226–1238 (2005)
10. Gómez, W., Pereira, W.C.A., Infantosi, A.: Improving classification performance of breast lesions on ultrasonography. Pattern Recogn. **48**(4), 1125–1136 (2015)
11. Gómez, W., Leija, L., Alvarenga, A.V., Infantosi, A.F.C., Pereira, W.C.A.: Computerized lesion segmentation of breast ultrasound based on marker-controlled watershed transformation. Med. Phys. **37**(1), 82–95 (2010)
12. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. Inf. Process. Manage. **45**(4), 427–437 (2009)