Toolbox for Ear Biometric Recognition Evaluation

Conference Paper · September 2015

DOI: 10.1109/EUROCON.2015.7313758

2 authors:

Žiga Emeršič
University of Ljubljana

Peter Peer
University of Ljubljana

Toolbox for Ear Biometric Recognition Evaluation

Žiga Emeršič, Peter Peer
Computer Vision Laboratory, Faculty of Computer and Information Science, University of Ljubljana
Večna pot 113, SI-1000 Ljubljana, Slovenia
Email: {ziga.emersic, peter.peer}@fri.uni-lj.si

***Abstract*—Ears are not subjected to facial expressions like faces are and do not require closer inspection like fingerprints do. However, there is a problem of occlusion, different lightning conditions and angles. These properties mean that the final outcome depends heavily on the selected database and classification procedures used in the evaluation process. Moreover, the results metrics are often difficult to compare, different sections of evaluation procedure mask the important steps, and frameworks that are usually build on-the-fly take time to develop. With our toolbox we propose the solution to those problems enabling faster development in the field of ear biometric recognition.**

I. Introduction

In recent years many studies have been made in the field of person recognition using ear biometric data. While many of them present interesting results and usable solutions it is often difficult to directly compare them. Experiments are performed using different databases and custom classifiers with different evaluation matrices inside individually developed frameworks. All of this brings variability into the evaluation process and masks the performance of ear features detection, ear feature descriptors and ear recognition.

When preparing good recognition system, classifiers are usually optimized or developed specifically for the problem. While this is good when considering the system as a whole, it masks the information regarding the actual performance of ear feature extractors. And when considering the field of computer vision as such and the field of person recognition based on ear biometric data, this is the part on which the emphasis should be put.

Databases that are taken under supervised conditions often lack the actual difficulties that appear in the wild. The solution we propose is to use in the wild database with fixed classifiers. In this way the performance comparison falls directly onto the ear descriptors. Then there is also a problem that researchers usually need to develop their own framework in which experiments are conducted. This takes effort, time and presents yet another risk of possible errors and unwanted variability. With our CVL (Computer Vision Laboratory, Faculty of Computer and Information Science, University of Ljubljana) ear toolbox we try to minimize these issues and ease the process of evaluation in the field of ear biometric recognition, thus enabling faster development in the field.

Let us note that there are two types of biometric recognition systems: verification and identification. Although the toolbox presented in this paper supports both types, we focused on verification during the experiments. Our main goal was to present the toolbox and how it works and not to optimize certain biometric methods or to focus specifically on verification (or identification for that matter).

In Section II related tools and existing ear databases are presented. The following Section III presents CVL ear toolbox and how to use it. In Section IV experiments that we conducted using CVL ear toolbox are described to prove its usability. Section V gives conclusion and plans for future work.

978-1-4799-8569-2/15/$31.00

II. Related work

A. Evaluation systems

Biometric evaluation tools are available, but they are either intended for general biometric use [1] [2] or specified to use with i.e. face [3] [4]. Currently there are no free and widely used tools specifically designed for ear biometrics data available.

From a general perspective the Biometrics Evaluation and Testing (BEAT) [2] platform is interesting. BEAT provides an on-line open platform for conducting biometric experiments. Its goal is to propose a framework of standard operational evaluations for biometric technologies. It is comprehensive and provides framework for different levels of experiments, data presentation and the repeatability of experiments, but the problem is that it does not provide an ear database in the wild and is available only on-line. The problem of evaluation methodology in biometrics have also been addressed in [5], [6], [7]. In [8] the authors addressed the problem of biometric systems' ease of use and vulnerabilities instead of focusing only on the performance as it is usually the case. This however, was not our focus when designing CVL ear toolbox.

B. Databases

In the experiments we compared WPUTEDB and IIT Delhi ear database with our CVL ear database that is part of our toolbox. We chose these two databases due to their properties and the fact that both were created in a controlled environment: WPUTEDB, similarly to CVL ear database, contains color images with occlusions, varying angles with both left and right ears, the IIT Delhi ear database contains grayscale images of left ears without significant occlusions.

WPUTEDB – The WPUTEDB database contains 3348 images of 421 subjects [9] with 4 to 10 images per subject. Images were taken under different indoor lightning conditions and different angles and contain left and right ears with occlusions.

IIT Delhi ear database – The IIT Delhi ear database contains 493 grayscale images of 125 subjects [10] with 3 to 6 images per subject. Images were taken under different indoor lightning conditions with a fixed profile angle and contain right ears with no occlusions.

University of Notre Dame databases – The University of Notre Dame databases are composed of multiple dataset collections: three collections of combined 3480 images of 952 subjects (averaging 3 to 4 images per subject) contain both 3D and 2D profile ear images and a collection of 2D-only dataset collection containing 464 images from 114 subjects [11] [12] with 3 to 9 images per subject. Images were taken under different lightning conditions and angles and contain left ears only.

UBEAR dataset – The UBEAR dataset contains 4429 images from 126 subjects with the average of 35 images per subject. Images were taken under varying lightning condition and varying angles and contain left and right ears with occlusions [12], [13].

More about CVL ear database is given in the next section.

III. CVL EAR TOOLBOX

The main issues with the current evaluation practices in the field of ear biometrics that we found are: the use of different databases, different evaluation matrices, different classifiers that mask the feature extraction performance, and the time spent developing framework. To overcome these issues CVL ear toolbox was developed.

The CVL ear toolbox is implemented in Matlab and is (same as the CVL ear database) freely available per request. It contains needed wrapper functions as well as CVL ear database.

A. Toolbox outline

The toolbox provides environment in which the evaluation of methods for person recognition based on ear biometric data is simplified. It executes all of the database reads and classification based on ear descriptors. The only task of the researchers using the toolbox is the Matlab implementation (or C/C++ using Mex) of a function, which receives ear image data and outputs vector of ear descriptors. For each vector it is desired that it is of the same size, but it is not necessary. In the latter case each vector should be normalized in the postprocessing step.

It is also possible that researchers provide their own classifiers. In that case `learn` and `predict` functions need to be overwritten. The researchers can also use completely independent classification procedures (without the toolbox) since the extracted feature vectors are stored in plain text files (which also enables later re-evaluation). In Figure 1 the toolbox's functioning is visualized. In Figure 2 the backbone of the toolbox is shown. The toolbox in the evaluation process uses random sub-sampling validation by default. This validation is useful because the train-test division ratios do not need to be divisible by the number of iterations. The number of iterations and the ratio between test and train set can be changed through the parameters and are not mutually dependent.

Five segments are available for implementation and/or tuning, including a group of general settings that tune the toolbox,

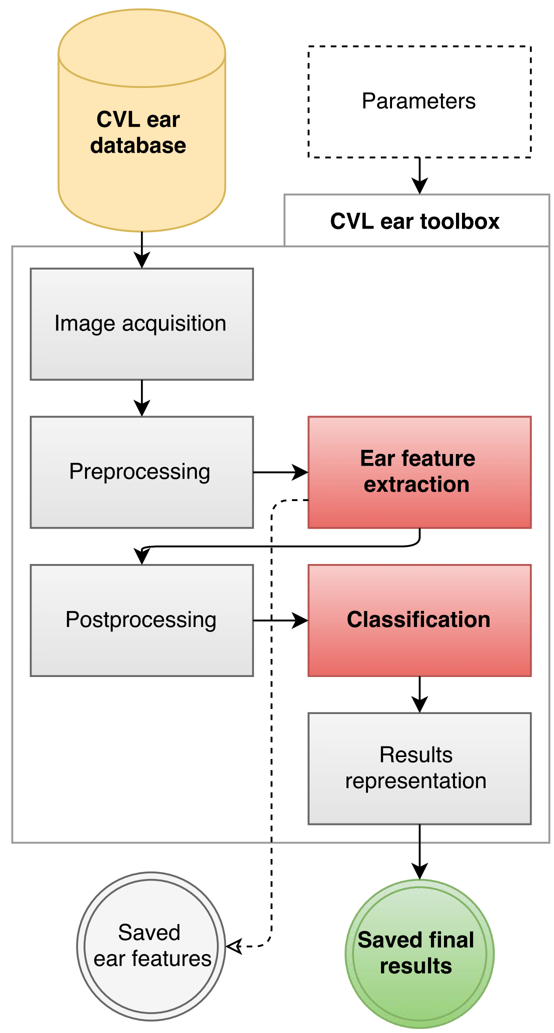

Fig. 1: Diagram showing the CVL ear toolbox evaluation process

```
% READ DB
% the default path links to
% the included CVL ear database
[db, annotation_data] = database(path);

% PREPROCESSING
db = preprocess(db, annotation_data);

% FEATURE EXTRACTION
% features_extract(image, image_annotation_data)
% is called inside on each db image
features = features_extract_all(db, annotation_data);

% POSTPROCESSING
features = postprocess(features);

% EVALUATION
% learn(X, y, annotation_data)
% and predict(model, X, annotation_data)
% are called inside for each cross-validation step
results = evaluate(features, annotation_data);

% OUTPUT THE RESULTS
visualize_results(results)
```

Fig. 2: The backbone of CVL ear toolbox

```
function [db] = preprocess(db, annotations)
 % preprocess applies preprocessing to
 % each image in db
 %
 % Input:
 %    db           = all ear images
 %    annotations = annotation data for
 %                  each image in db
 %
 % Output:
 %    db           =  modified images
```

Fig. 3: The optional preprocessing function

```
function [features] = features_extract
                       (image, annotation)
 % extract ear features for current image
 %
 % Input:
 %    image       = already preprocessed
 %                  ear images
 %    annotation = annotation data for
 %                  this image
 %
 % Output:
 %    features    = vector of ear features
```

Fig. 4: The mandatory features extraction function

they are described in the following paragraphs. Accompanying figures give only the high-level calls and functions signatures with explanations. They are given in Matlab as the paper also serves as the basic toolbox usage tutorial, but can also be viewed as pseudocode.

The preprocessing function (optional) – The function in Figure 3 is optional and if the code is not present the toolbox will skip this step. The main focus of this step is the option to normalize or to modify input data (ear images) with the goal of increasing performance of feature extraction and classification.

The ear feature extraction function (mandatory) – Implementation of this function is mandatory for the process of evaluation to be completed. Its signature is shown in Figure 4. Some example functions are already present in the toolbox. If the code is not present, toolbox will stop the evaluation. In the function the user needs to implement the feature extraction procedure and then output the ear descriptor in a 1-dimensional vector of features. As the input the function receives an ear image and the corresponding annotation data. With the default CVL ear database these are tragus location, image dimensions and ear direction.

Note that subject ID is not a part of this group of annotation data and the `feature_extract` function does not receive any information regarding the class affiliation, the id is given in a form of vector `y` to the `learn` function, shown in Figure 6. The toolbox takes care of all the looping and reading from the database. The output is then fed into the next step, where classification training and testing is performed.

In our experiments we have used Histograms of Oriented Gradients (HOG) [14], Speeded Up Robust Features (SURF) [15], Maximally Stable Extremal Regions (MSER) [16] and Scale-Invariant Feature Transform (SIFT) [17] for ear descriptors acquisition. They are described in Section IV.

The postprocessing function (optional) – This function takes as an input an array of all ear description vectors. The purpose of this function is to modify ear descriptors if needed, i.e. reshape, filter descriptors. If no custom code is provided the toolbox will skip this step. The signature of the function is shown in Figure 5.

```
function [features] = postprocess(features)
 % do postprocessing on all features
 %
 % Input:
 %    features = array of ear features
 %               for each image in db
 %
 % Output:
 %    features = modified features
```

Fig. 5: The optional postprocessing function

```
function [model] = learn(X, y, annotations)
 % classifiers fitting/learning is
 % performed
 %
 % Input:
 %    X            = features of the
 %                   training set
 %    y            = ground truth (classes)
 %    annotations = annotation data for
 %                   each example in X
 %
 % Output:
 %    model = learned classifier

function [y] = predict(model, X, annotations)
 % make a prediction
 %
 % Input:
 %    model        = learned classifier
 %    X            = features of the
 %                   test set
 %    annotations = annotation data for
 %                   each example in X
 %
 % Output:
 %    y            = predicted classes
```

Fig. 6: The optional classification functions

The classification function (optional) – The implementation of this function is optional and if the code is not present the toolbox will use the default SVM classifier. Tuning of parameters is not available for the existing classifier, but the researches should feel free to copy the code of the internally used classifier and use it inside their classification function override. The default behavior could be overridden using `learn` and `predict` functions.

The signatures of `learn` and `predict` functions are shown in Figure 6. Both functions receive an array of ear descriptor vectors as the input. In the `learn` function class affiliation is also given. The latter is needed for classifier to learn. This function returns the learned model that is then used during prediction phase (`predict` function). The calls of these two functions are done by the toolbox automatically during evaluation (cross-validation) procedure.

The toolbox parameters (optional) – This part consists of general parameters that the toolbox then uses during evaluation process. The parameters include information whether to use internal CVL ear database or a custom one. The definition of the output results format and representation, format and location of input annotation data (if users decides to use different database), number of evaluation runs, identification/verification mode, train/test set ratio etc. Parameters are initially set so that the toolbox works out of the box.

B. CVL ear database

When images are captured under supervised conditions (e.g. in a laboratory) all images share certain properties that would otherwise not be found in the wild. Images are often taken using the same imaging tools, under similar lightning conditions, within short span of time (days or mostly weeks). When experiments are repeated on in the wild database or on other database, unpredicted difficulties can arise. To overcome this it is important to use a database that is sufficiently challenging and as similar to the final environment as possible. We propose that the best database is the one consisting of the images taken in the wild.


(a) Significant angle


(b) Image in grayscale


(c) Light occlusion by hair


(d) Heavy occlusion by hair


(e) Heavy occlusion by hair with an earring

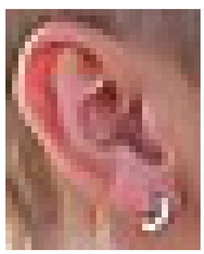
(f) Low resolution image

Fig. 7: CLV ear database representative samples

In CVL ear database that is a part of the presented CVL ear toolbox the presented issues were addressed to enable a thorough evaluation of current and future ear recognition methods. However, researchers are able to use arbitrary database with the CVL ear toolbox as long as it follows the rules listed below. These rules are necessary for the toolbox to accept the database and performs evaluation tests correctly:

- Images must be divided by persons in folders – each person in a separate folder. The names of the folders are recommended to be simple numerical IDs.
- Images must be of the following formats: Portable Network Graphics (PNG), Joint Photographic Experts Group (JPEG) or bitmap (BMP).
- Other optional annotation data should be in JavaScript Object Notation (JSON) format. One file per subject (folder) and stored inside the folder with the file ending ".json".

The CVL ear database currently consists of 804 images of 16 well known subjects with images per subject from 19

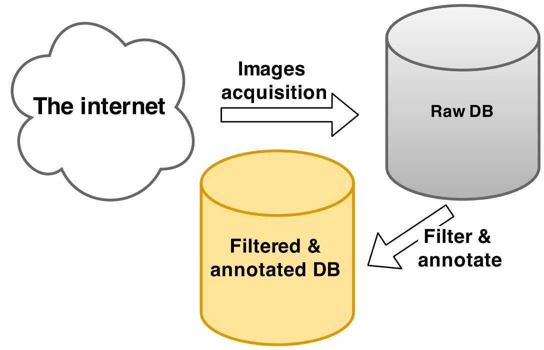

Fig. 8: Diagram showing CVL ear database creation process

to 94 and is freely available per request. Images were taken indoor and outdoor under different lightning conditions as is usually the case with the images in the wild. Images in the database vary in size and quality and are stored in PNG format. Majority of the images are under 200×200 pixels in size (94% of them). Images were also taken at different angles ranging from complete profile to complete frontal images. There are no restrictions regarding occlusions (hair, earrings, earphones etc. are all present in the images and some examples can be seen in Figure 7) or time differences between acquisitions of images (images of subjects were taken at different times and eras; we estimate that the difference in some of the images are up to 30 years) – something that we were not able to find among existing databases.

The CVL ear database was created in a semi-automatic fashion. The creation process consisted of two main steps as shown in Figure 8: image acquisition from the Internet and image preparation that includes selection of images and annotation. The last step – image annotation with cropping was done manually, because as high precision as possible was needed. [18]

Annotation data is stored in a JSON format, but if needed can be transformed into an arbitrary format. Each corresponding annotation data consist of image dimensions, a tragus center point, ear direction and a file name. The center point is the location of the outer area of the tragus.

It is important to notice that images uploaded on the Internet are often mirrored – during image acquisition we noticed that sometimes background behind subjects indicate that the image is mirrored with various labels and captions being mirrored. This means the image direction attribute describes the direction in which the center of the ear is; and not necessarily whether the ear is from the left or from the right side of the head. Taking this fact into account, we believe that ear direction could prove to be useful parameter in the future. The majority of cases still indicate, which ear is in the image.

Tragus point could be proven useful when using concentric circles as holistic descriptors [12] or when distance between tragus and outline of the ear is used [19].

In the case of recognition system in the wild it would be on authors to calculate the tragus center point if needed and whether the ear is on the left or on the right side of a head in real time. The same goes for the ear detection itself. Our goal is to provide a foundation for a faster and better development of ear recognition techniques as such and not also for ear detection.

C. Output

The toolbox at the end of evaluation returns the overall accuracy (performance), the number of correctly classified positives and negatives (true positives TP, true negatives TN), the number of wrongly classified positives and negatives (false positives FP, false negatives FN), the number of all real positives and negatives, the sensitivity, the specificity, the receiver operating characteristic, the area under the (receiver operating characteristic) curve and the equal error rate.

IV. Evaluation experiment

To evaluate CVL toolbox itself, we decided to use four different ear feature descriptors, i.e. Histogram of Oriented Gradients (HOG) [14], Speeded Up Robust Features (SURF) [15], Maximally Stable Extremal Regions (MSER) [16] and SIFT Scale-Invariant Feature Transform (SIFT) [17], and three different databases. As data input we used different databases to show how CVL ear database compares to the existing ones. We did not focus on evaluating HOG, SURF, MSER or SIFT but rather the difference in classification (verification) between different databases and how this is done using CVL ear toolbox.

Due to the nature of the ear images in CVL ear database (taken in the wild) the performance on it is expected to be worse than on existing databases, because images in the wild are generally of lower overall quality and not taken under controlled conditions. The deviation nevertheless should not be too big, even when using basic methods, because that would mean that the database is either too difficult to use or that images are of insufficient quality for use in ear biometrics. The experiments therefore verify the applicability of the CVL ear toolbox's default database – CVL ear database.

The first step in the toolbox is the database loading into the Matlab environment. We conducted used CVL ear database that is included in the CVL ear toolbox (it represents a baseline database), WPUTEDB [9] and IITDelhi ear database [10].

The second step in the CVL ear toolbox is preprocessing. In all cases (using HOG, SURF, MSER and SIFT) we transformed images to grayscale. In the case of HOG we additionally transformed images to fixed dimensions of 100×100 pixels.

SIFT is a local method that is invariant to image scaling rotation and partially to change in illumination and 3D transformation [17]. The mentioned properties make SIFT a good method for handling ear biometric data. It consists of four major steps: scale-space extrema detection, key point localization, orientation assignment, key point description [17], [20]. We added the fifth in the postprocessing step – k-means clustering for dimension reduction.

The HOG method is useful when illumination variations or shadowing are present [14], [21]. The first step in the HOG calculation process (after image color transformation to grayscale) was image size transformation to fixed 100×100 pixels. Since HOG returns fixed number of features for a given image size this enables us to feed the results directly into SVM without applying k-means clustering (or other method) first.

SURF is a local method, inspired by SIFT, though faster [15]. This is achieved by relying on integral images for image convolutions and using principles of the Hessian matrix-based measure for the detector and distribution-based descriptors [15]. In the postprocessing step we used k-means clustering for dimension reduction.

MSER is a method invariant to affine transformation in images and different illumination conditions [16], which makes it useful in the field of ear recognition. It uses the so-called extremal regions in images as a basis for the image comparison. As in the cases of SIFT and SURF, we used k-means clustering for dimension reduction in the postprocessing step of evaluation process.

The main reason for using k-means clustering in the postprocessing step is that the input vectors into the last, classification step need to be of the same size.

In the last step (classification) Support Vector Machine was used to perform person verification.

The results are presented in Table I. During learning and testing processes we did not differentiate between left or right ears, because according to [11] 90% of people's right and left ear are symmetric. Performance evaluation was done using the toolbox's default mode: repeated random sub-sampling validation with three runs. Database was randomly divided into test and train groups with ratios of 3 to 7 and repeated three times. The final score is the average over all runs with the standard deviation of less than 0.02 for the overall performance.

Performance is the overall ratio of correctly classified subject vs. all subjects and defined as $\frac{T}{F+T}$, where F and T represent a number of falsely and correctly classified examples, respectively. Specificity is defined as $\frac{N_T}{N}$, where N_T represents true negatives (correctly classified as negatives) and N all negatives. Sensitivity is defined as $\frac{P_T}{P}$, where P_T represents true positives and P all positives. AUC or Area Under the Curve represents the area under the Receiver Operating Characteristic (ROC) curve. Values of AUC can range from 50% (random classification, useless) to 100% (perfect classification).

The experiments show that classification on CVL ear database generally performs slightly worse than on other two: 97.76% with differences of 1.86 percent points (pp) and $2.15pp$ using HOG, 92.36% with differences of $7.19pp$ and $7.19pp$ using SURF, 91.35% with differences of $7.98pp$ and $8.23pp$ using MSER, 88.91% with differences of $10.58pp$ and $10.94pp$ using SIFT.

The reason for deviation where performances of SURF, MSER and SIFT drop when using CVL ear database is, we presume, that k-means clustering (which was not used with HOG) reduces ear feature descriptors too much.

Images in CVL ear database are smaller than images in other two and so are the feature descriptors. When we apply k-means clustering on the already small vectors too much information is lost and performance drops. This is especially noticeable when sensitivity and AUC are observed. When the overall performance reduces, sensitivity and AUC will be more influenced because there are a lot more negative samples than the positive ones.

The fact that classification on CLV ear database performs worse than on other two databases is as expected as it is in the wild database, but at the same time the performance is not much worse – if it would be this would mean that the database is not applicable when widely used methods for feature extraction like HOG, SURF, MSER or SIFT are used. This shows that CVL ear database provides important and challenging source of data for ear biometric experiments. And the publicly available CVL ear toolbox gives a good foundation for easier evaluation and comparison of ear biometric recognition methods.

TABLE I: Results using HOG, SIFT, SURF and MSER descriptors with SVM classifier

[%]	CVL ear database	IIT Delhi ear database	WPUTEDB
HOG			
Performance	97.76	99.62	99.91
Specificity	99.75	100.00	100.00
Sensitivity	65.78	51.75	57.13
AUC	82.76	75.87	78.56
SIFT			
Performance	88.91	99.49	99.85
Specificity	93.28	99.90	99.94
Sensitivity	17.94	46.81	58.16
AUC	55.61	73.36	79.05
SURF			
Performance	92.40	99.55	99.56
Specificity	98.30	99.90	99.70
Sensitivity	6.48	54.68	29.85
AUC	52.39	77.29	64.78
MSER			
Performance	91.35	99.33	99.58
Specificity	97.18	99.72	99.74
Sensitivity	11.49	49.65	22.76
AUC	54.34	74.69	61.25

V. Conclusion

We addressed the difficulties in evaluation of ear biometric identification methods and presented the first publicly available ear biometric toolbox. The experiments were done using different databases and four different feature extraction methods to demonstrate the use of the toolbox. We believe that this toolbox can provide faster implementation and standardized evaluation of new ear biometric recognition methods and contribute to faster development in the field.

Our plan is to upgrade the toolbox with better results visualization, to expand the database, to add additional ready to use functions into the toolbox and to respond to community needs. We are also developing a web interface that will enable upload of CVL ear toolbox's output and provide graphical visualizations of the results, together with the on-line comparisons to other researchers' results.

References

[1] R. Schultz and R. Ives, "Biometric data acquisition using matlab guis," in *Frontiers in Education, FIE. Proceedings 35th Annual Conference*, Oct 2005.

[2] I. R. Institute. (2015) Biometrics evaluation and testing (BEAT). [Online]. Available: https://www.beat-eu.org/

[3] V. Štruc. (2009) Inface: A toolbox for illumination invariant face recognition. [Online]. Available: "http://luks.fe.uni-lj.si/sl/osebje/vitomir/face_tools/INFace"

[4] G. Littlewort, J. Whitehill, T. Wu, I. Fasel, M. Frank, J. Movellan, and M. Bartlett, "The computer expression recognition toolbox (cert)," in *Automatic Face and Gesture Recognition and Workshops (FG), 2011 IEEE International Conference on*, March 2011, pp. 298–305.

[5] D. Gorodnichy, "Evolution and evaluation of biometric systems," in *Computational Intelligence for Security and Defense Applications, CISDA. IEEE Symposium on*, July 2009, pp. 1–8.

[6] L. Omelina and M. Oravec, "Universal biometric evaluation system: Framework for testing evaluation and comparison of biometric methods," in *Systems, Signals and Image Processing (IWSSIP), 18th International Conference on*, June 2011, pp. 1–4.

[7] J. Lacirignola, P. Pomianowski, D. Ricke, D. Strom, and E. Wack, "Multimodal biometric collection and evaluation architecture," in *Homeland Security (HST), IEEE Conference on Technologies for*, Nov 2012, pp. 61–66.

[8] B. Fernandez-Saavedra, R. Alonso-Moreno, J. Uriarte-Antonio, and R. Sanchez-Reillo, "Evaluation methodology for analyzing usability factors in biometrics," in *Security Technology, 43rd Annual International Carnahan Conference on*, Oct 2009, pp. 347–354.

[9] D. Frejlichowski and N. Tyszkiewicz, "The west pomeranian university of technology ear database – a tool for testing biometric algorithms," in *Image Analysis and Recognition*, ser. Lecture Notes in Computer Science, A. Campilho and M. Kamel, Eds. Springer Berlin Heidelberg, 2010, vol. 6112, pp. 227–234. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-13775-4_23

[10] A. Kumar and C. Wu, "Automated human identification using ear imaging," *Pattern Recogn.*, vol. 45, no. 3, pp. 956–968, Mar. 2012. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2011.06.005

[11] P. Yan and K. Bowyer, "Empirical evaluation of advanced ear biometrics," in *Computer Vision and Pattern Recognition – Workshops (CVPR), IEEE Computer Society Conference on*, June 2005, pp. 41–41.

[12] A. Pflug and C. Busch, "Ear biometrics: a survey of detection, feature extraction and recognition methods," *Biometrics IET*, vol. 1, no. 2, pp. 114–129, June 2012.

[13] R. Raposo, E. Hoyle, A. Peixinho, and H. Proenca, "UBEAR: A dataset of ear images captured on-the-move in uncontrolled conditions," in *Computational Intelligence in Biometrics and Identity Management (CIBIM), IEEE Workshop on*, April 2011, pp. 84–90.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Conference on*, vol. 1, June 2005, pp. 886–893.

[15] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer Vision ECCV*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer Berlin Heidelberg, 2006, vol. 3951, pp. 404–417. [Online]. Available: http://dx.doi.org/10.1007/11744023_32

[16] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885604000435

[17] D. Lowe, "Object recognition from local scale-invariant features," in *Computer Vision, IEEE International Conference on*, vol. 2, 1999, pp. 1150–1157.

[18] Ž. Emeršič and P. Peer, "Ear biometric database in the wild," in *Bio-inspired Intelligence (IWOBI), IEEE International Work Conference on*, June 2015, pp. 25–30.

[19] L. Lu, Z. Xiaoxun, Z. Youdong, and J. Yunde, "Ear recognition based on statistical shape model," in *Innovative Computing, Information and Control, ICICIC. First International Conference on*, vol. 3, Aug 2006, pp. 353–356.

[20] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. [Online]. Available: http://dx.doi.org/10.1023/B%3AVISI.0000029664.99615.94

[21] N. Damer and B. Fuhrer, "Ear recognition using multi-scale histogram of oriented gradients," in *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), International Conference on*, July 2012, pp. 21–24.