Open Source Presentation Attack Detection Baseline for Iris Recognition

Joseph McGrath      Kevin W. Bowyer      Adam Czajka
Department of Computer Science and Engineering, University of Notre Dame
{jmcgrat3,kwb,aczajka}@nd.edu

Abstract

This paper proposes the first, known to us, open source presentation attack detection (PAD) solution to distinguish between authentic iris images (possibly wearing clear contact lenses) and irises with textured contact lenses. This software can serve as a baseline in various PAD evaluations, and also as an open-source platform with an up-to-date reference method for iris PAD. The software is written in C++ and uses only open source resources, such as OpenCV. The method does not require iris image segmentation and uses Binary Statistical Image Features (BSIF) to extract PAD-related features, which are classified by an ensemble of SVM classifiers. The SVM models attached to the current software have been trained with the NDCLD'15 database and the correct recognition rate exceeds 98%. However, the software implements the functionality of retraining the classifiers with any database of authentic and attack images.

1. Introduction

Presentation attack detection (PAD) is an important element of biometric systems. There were multiple demonstrations of successful presentation attacks on commercial systems suggesting that the PAD mechanisms were either ineffective or missing in these systems, and iris recognition is not an exception. Starting from the first results in 2002 demonstrating that paper printouts can be matched to real irises [16] by a commercial system, and running through last year’s spoofing of iris recognition in the Samsung Galaxy S8 [1], we can conclude that iris PAD is not a solved problem. The most recent LivDet-Iris 2017 evaluation [21] additionally suggests that the open-set regime, in which some (or all) properties of samples are unknown during training, is even more challenging, as the winning algorithm did not recognize from 11% to 38% of attack images, depending on the database.

Iris PAD is a very dynamic research area, with many PAD algorithms proposed to date [3]. The question arises: which factors prevent us, as a community, from moving forward with making iris PAD more effective, especially for unknown attack types? One possible reason is the lack of an open source platform to maintain a baseline iris PAD methodology that is easy to contribute to and easy to benefit from when developing or evaluating original solutions. The OpenCV platform is a great example of such an initiative in computer vision in general. The Masek implementation and more recently the OSIris system have played this role for iris recognition. This paper plants a seed and offers the first open source software based on one of the more recent, effective iris PAD methods employing Binary Statistical Image Features (BSIF) and ensemble classification realized by Support Vector Machines (SVM) [4]. The initial version proposed in this paper includes the SVM-based ensemble already trained on one of the publicly available datasets of images of irises with and without textured contact lenses, NDCLD’15 [3], and hence is “ready to use”. However, one of the functionalities of this software is to retrain the ensemble with any samples, especially those which conform to the ISO/IEC 19794-6 standard. The proposed solution also delivers raw BSIF-based PAD features for those who want test other classifiers and ensembles. To our knowledge, this is the first and only proposal of an open source solution for iris presentation attack detection. It is specialized (in this current version) to detection of textured contact lenses and based on a strong, recent methodology of textured contact lens detection. The GitHub repository can be accessed at https://github.com/CVRL/OpenSourceIrisPAD.

2. Related Work

The number of iris PAD methods developed to date is significant, and a recent survey by Czajka and Bowyer [3] categorizes them into groups of methods using either still iris images or iris videos, which are either acquired passively (with no eye stimuli) or actively (when the eye is open).
stimulated by external light, or a response is expected from the subject). The group of methods using still samples in PAD, identical to those used in iris recognition, is mostly populated by solutions employing various texture descriptors (such as BSIF [11]), or – recently popular – convolutional neural networks [1]. If some modifications in the iris recognition equipment are possible, the iris PAD methods incorporate multi-spectral imaging solely in near-infrared band [13] or combined with visible-light imaging [17]. 3D properties of the eye [12], or dynamic features such as spontaneous [19] or stimulated [2] pupil oscillations, eye blinks [15], or eyeball movements [10]. Despite the large number of proposed PAD methods to date, Czajka and Bowyer [3] conclude that they “do not know of even a single well-populated method of iris PAD that is available to the research community as open source,” which motivated our publishing of this first open-source solution.

In the context of the existing tools for and efforts towards faster development of iris PAD methodologies, it is worth mentioning numerous benchmark databases, such as Clarkson, Warsaw, Notre Dame, and WVU/IIITD-Delhi developed for LivDet-Iris competitions [22, 23, 21] (paper printouts and textured contact lenses), NDCLD 2012 [7], NDCLD 2013, [6] and NDCLD 2015 [4] (clean and textured contact lenses), ATVS-FIr [8] (paper printouts), Pupil-Dynamics [2] (pupil size in time with and without visible-light stimuli), Post-Mortem-Iris [18] (images of irises acquired up to one month after death), CASIA-Iris-Syn [20] (synthetically generated iris images), or data acquired by a LightField sensor GUC-LF-VIAr-DB [14]. The LivDet-Iris competitions⁴, mentioned earlier, are an important effort towards independent evaluation of iris PAD algorithms. Editions were organized in 2013 [22], 2015 [23], and 2017 [21] and brought together researchers from around the world, who submitted their iris PAD algorithms for evaluation.

It is also worth mentioning the ISO PAD-related standardization efforts. In particular, the ISO/IEC 30107-1 standard defines the PAD framework and vocabulary, and is freely available⁵. The third part, ISO/IEC 30107-3 defining the PAD evaluation, has been adopted to the National Voluntary Laboratory Accreditation Program (NVLAP) run by NIST⁶.

We hope that the proposed open-source iris PAD software will fill the current gap of strong, recent open-source baseline iris PAD algorithms, stimulate their development, and see multiple contributors.

3. The Baseline Method for Textured Contact Lens Detection

The implemented solution follows the methodology proposed by Doyle and Bowyer [4] and the feature extraction is based on Binary Statistical Image Features (BSIF) proposed by Kannala and Rahtu [9]. In this method, the calculated “BSIF code” is based on filtering the image with n filters of size s × s, and then binarizing the filtering results with a threshold at zero. Hence, for each pixel n binary responses are given, which are in the next step translated into a n-bit grayscale value. In the original paper, n ∈ {5, 6, 7, 8, 9, 10, 11, 12}, and s ∈ {3, 5, 7, 9, 11, 13, 15, 17}, and thus there are 60 combinations of n and s (4 combinations, namely n ∈ {9, 10, 11, 12} for s = 3, were skipped). The filters, for each considered combination of n and s, were trained on patches extracted from natural images in a way to maximize the statistical independence of filter responses. Fig. 1 presents BSIF codes for example iris images (with and without textured contact lenses) for two example scales (s = 7 and s = 17) and n = 8.

The histograms resulting from gray-scale BSIF codes are later used as texture descriptors, and the number of histogram bins equals 2^n, as shown in Fig. 1. Following Doyle and Bowyer [4], in this implementation we use n = 8 only and, in addition to the original ISO-compliant iris image resolution of 640 × 480, we extract BSIF codes for an image down-sampled to 320 × 240. This allows for the exploration of more scales in feature extraction. Consequently, we end up with 16 histograms for each image, and a separate SVM is trained for each feature set. Since not all the classifiers have the same strength, a subset of the strongest classifiers is selected and majority voting is applied to come up with a final decision. The set of strongest classifiers can be configured in the proposed solution.

4. Software Architecture

The TCL Detection solution proposed in this paper was written in C++ and tested on MacOS High Sierra 10.13.6 using g++ 4.2.1 as the compiler, and on Windows 7 64 bit using Microsoft Visual Studio 2015. The implementation depends on many of the functions available in OpenCV; specifically, version 3.4.1 was used for this release. TCL Detection is organized into three main modes of operation: feature extraction, model training, and model testing.

The architecture of the program is based upon OSIRIS version 4.1, which can be seen through the use of the TCLManager class to handle the information flow within the program. The manager contains methods to parse the configuration file, show the configuration, and run the desired mode of operation.

---

⁴http://livdet.org
⁵https://goo.gl/JSbiqy
⁶https://www.nist.gov/nvlap
Figure 1: 8-bit BSIF codes and the resulting histograms calculated at two example scales for an authentic iris image and an iris image with textured contact lens.

Figure 2: The schematic view of the proposed open-source iris PAD solution.

The featureExtractor class handles the extraction of BSIF features through an extract() method. Within the extractor, a private method filter() creates an instance of the BSIF filter that is then used to filter a regular and down-sampled version of each image. The BSIFFilter class contains the methods for loading the hard-coded filters and for generating a histogram when given an image. The scales included in BSIF are $s = \{3, 5, 7, 9, 11, 13, 15, 17\}$, as mentioned above, but using a down-sampled image essentially doubles the filter size, allowing $s = \{6, 10, 14, 18, 22, 26, 30, 34\}$ to be used as well. Each BSIF scale also has a bit size parameter $n$; TCL Detection has been tested for $n = 8$, but any $n \in \{5, 6, 7, 8, 9, 10, 11, 12\}$ can be used, if needed. The response of each location within an image to the filter is assigned a bit in an 8-bit integer and a histogram of the image response is taken as the feature space. Once the features are calculated, they are output to comma-separated value files for future use.

The remainder of the operation modes are handled by the manager class, which instantiates the required OpenCV objects to train models and test images. If model training is enabled, the manager will call a method to load the features for the images specified in the training list. The training features and classifications are then loaded into an instance of the TrainData class from OpenCV. A new SVM with
the kernel set to the radial basis function is then initialized and trained. Training is achieved through the `trainAuto` function in OpenCV: this function chooses the optimal parameters for the SVM using k-fold cross validation with ten folds. Each model that is trained is then output as an XML file.

If model testing is enabled, the manager will load all required models from their XML files. Once all models are loaded, the manager will determine whether majority voting is enabled. If majority voting is disabled, the manager will select each model individually and load the testing features corresponding to the BSIF scale the model was trained on. These features will then be input to the predict function for the model from OpenCV and the predictions will be returned. The predictions will be tested against the classifications provided with the test set and the accuracy will be output for each model individually.

If majority voting is enabled, the predictions for each model are determined and temporarily stored. For each image, the number of models voting for each classification is determined and the overall decision is made with a simple majority vote. In the case of a tie, a random decision is made. The ensemble accuracy on the training set is then determined through comparison with the classifications provided with the test set.

5. Results

5.1. Novel Textured Lens

To evaluate the effect of a novel lens type, an ensemble of SVMs was trained on textured lenses from four manufacturers and tested on a verification set containing textured lenses from the fifth. The images used were from a mix of sensors – LG2200, Iris Guard AD 100, and LG 4000 – and the whole image was used during feature extraction. Models were trained on a set consisting of 1000 clear lens or no lens images and 1000 textured lens images, 250 of each of four brands. The models were tested on a set consisting of 250 images of the remaining brand and 250 clear or no lens images. For both sets, the clear and no lens images were subject disjoint and randomly selected from a larger pool of subjects. The pool of subjects with textured lenses was too small to ensure that the textured lens images were subject disjoint. This, however, is not that critical since the textured contact lenses cover almost the entire annulus of actual iris texture, so having the same subjects presenting their irises with different textured lenses is similar to a situation of having different subjects presenting these textured lenses.

The correct classification rate (CCR) on the testing set, which contained images from the fifth brand, was taken as the performance measure. Sixteen models were trained on each of five permutations, giving 80 models total. The average performance on the testing set across all models was 85.8%. This result is less than that achieved in [5] but not entirely unexpected as no segmentation was used in this result while best guess segmentation was used in the original paper.

Fig. 3 shows the performance of each BSIF scale across all five permutations of training and testing brands. It can be seen that the larger scales tend to generalize better than the smaller scales. Fig. 4 shows the performance of all BSIF scales by brand left out: CibaVision, Clearlab, CooperVision, Johnson&Johnson, and United Contact Lenses. All the results oscillate around 90%, except for CooperVision lenses, which have a different texture than the other four brands.

![Figure 3: Box plots presenting the performance of individual SVMs trained on BSIF features for different BSIF filter sizes across five leave-1-out permutations. Red bars denote median values, height of each boxes equals an interquartile range (IQR) spanning from the first (Q1) to the third (Q3) quartile, and the whiskers span from Q1-1.5*IQR to Q3+1.5*IQR. Outliers are shown as red crosses.](image)

5.2. Models for Release

To provide a complete solution in the open source release, sixteen SVMs were trained using the majority of the NDCLD’15 database, and representing all five contact lens brands, to increase the generalization capabilities of the solution. The 7,300 images in the database were divided into an 80:20 split, giving 5,840 images in the training set and 1,460 images in the validation set. These images were randomly selected from those available in the whole NDCLD’15 dataset, but were not subject disjoint.

After the SVMs were trained on the training set, each model was tested on the validation set to ensure that the models were behaving as expected and to provide an indication of the performance of each model. Figure 5 shows the individual performance of each model and indicates that the
larger BSIF scales provide better PAD features. This performance indication allowed the models to be added from best to worst to an ensemble of models. All ensemble sizes were able to achieve correct classification rates of around 98%, as can be seen in Figure 6.

6. Summary

This paper offers the first, known to us, open-source software solution for iris presentation attack detection. It is based on a recent and effective methodology of using Binary Statistical Image Features and an ensemble of classifiers to detect textured contact lenses. A trained ensemble of SVM classifiers, added to this initial version, achieves a correct classification rate of 98% on a popular NDCLD’15 benchmark, in a close-set scenario and without iris segmentation required. This software allows to retrain the ensemble with other datasets, define which SVM classifiers form the ensemble, and calculate BSIF-based features that can be used to test other classifiers worth adding to the ensemble. The long-term goal of this effort is to build an open-source baseline methodology for iris PAD, for instance for the next editions of the LivDet-Iris competition, starting from a recent and effective algorithm of textured contact lens detection.

References

[1] C. Chen and A. Ross. A multi-task convolutional neural network for joint iris detection and presentation attack detection. In IEEE Winter Conf. on Applications of Computer Vision (WACV), pages 44–51, March 2018.

[2] A. Czajka. Pupil dynamics for iris liveness detection. IEEE Trans. Inf. Forens. Security, 10(4):726–735, April 2015.

[3] A. Czajka and K. W. Bowyer. Presentation attack detection for iris recognition: An assessment of the state-of-the-art. ACM Computing Surveys, 51(4):86:1–86:35, July 2018.

[4] J. S. Doyle and K. W. Bowyer. Robust detection of textured contact lenses in iris recognition using bsif. IEEE Access, 3:1672–1683, 2015.

[5] J. S. Doyle and K. W. Bowyer. Robust detection of textured contact lenses in iris recognition using bsif. IEEE Access, 3:1672–1683, 2015.
[6] J. S. Doyle, K. W. Bowyer, and P. J. Flynn. Variation in accuracy of textured contact lens detection based on sensor and lens pattern. In IEEE Int. Conf. on Biometrics: Theory Applications and Systems (BTAS), pages 1–7, Arlington, VA, USA, September 2013. IEEE. 2

[7] J. S. Doyle, P. J. Flynn, and K. W. Bowyer. Automated classification of contact lens type in iris images. In 2013 Int. Conf. on Biometrics (ICB), pages 1–6, Madrid, Spain, June 2013. IEEE. 2

[8] J. Galbally, J. Ortiz-Lopez, J. Fierrez, and J. Ortega-Garcia. Iris liveness detection based on quality related features. In 2012 5th IAPR Int. Conf. on Biometrics (ICB), pages 271–276, New Delhi, India, March 2012. IEEE. 2

[9] J. Kannala and E. Rahtu. Bsif: Binarized statistical image features. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), pages 1363–1366, Nov 2012. 2

[10] O. V. Komogortsev, A. Karpov, and C. D. Holland. Attack of mechanical replicas: Liveness detection with eye movements. IEEE Trans. Inf. Forens. Security, 10(4):716–725, April 2015. 2

[11] J. Komulainen, A. Hadid, and M. Pietikäinen. Contact lens detection in iris images. In C. Rathgeb and C. Busch, editors, Iris and Periocular Biometric Recognition, chapter 12, pages 265–290. IET, London, UK, 2017. 2

[12] E. C. Lee and K. R. Park. Fake iris detection based on 3d structure of iris pattern. Int. Journal of Imaging Systems and Technology, 20(2):162–166, 2010. 2

[13] J. H. Park and M. G. Kang. Multispectral iris authentication system against counterfeit attack using gradient-based image fusion. Optical Engineering, 46(11):117003–117003–14, 2007. 2

[14] R. Raghavendra and C. Busch. Presentation attack detection on visible spectrum iris recognition by exploring inherent characteristics of light field camera. In IEEE Int. Joint Conf. on Biometrics (IJCB), pages 1–8, Clearwater, FL, USA, Sept 2014. IEEE. 2

[15] K. B. Raja, R. Raghavendra, and C. Busch. Video presentation attack detection in visible spectrum iris recognition using magnified phase information. IEEE Trans. Inf. Forens. Security, 10(10):2048–2056, October 2015. 2

[16] L. Thalheim, J. Krissler, and P.-M. Ziegler. Biometric Access Protection Devices and their Programs Put to the Test, Available online in c’t Magazine, No. 11/2002, p. 114. on-line, 2002. 1

[17] S. Thavalengal, T. Nedelcu, P. Bigioi, and P. Corcoran. Iris liveness detection for next generation smartphones. IEEE Trans. Cons. Elect., 62(2):95–102, May 2016. 2

[18] M. Trokielewicz, A. Czajka, and P. Maciejewicz. Human iris recognition in post-mortem subjects: Study and database. In IEEE Int. Conf. on Biometrics: Theory Applications and Systems (BTAS), pages 1–6, Niagara Falls, NY, USA, Sept 2016. IEEE. 2

[19] F. M. Villalbos-Castaldi and E. Suaste-Gómez. In the use of the spontaneous pupillary oscillations as a new biometric trait. In Int. Workshop on Biometrics and Forensics, pages 1–6, Valletta, Malta, March 2014. IEEE. 2

[20] Z. Wei, T. Tan, and Z. Sun. Synthesis of large realistic iris databases using patch-based sampling. In Int. Conf. on Pattern Recognition (ICPR), pages 1–4, Tampa, FL, USA, Dec 2008. IEEE. 2

[21] D. Yambay, B. Becker, N. Kohli, D. Yadav, A. Czajka, K. W. Bowyer, S. Schuckers, R. Singh, M. Vatsa, A. Noore, D. Gragnaniello, C. Sansone, L. Verdoliva, L. He, Y. Ru, H. Li, N. Liu, Z. Sun, and T. Tan. LivDet Iris 2017 – iris liveness detection competition 2017. In IEEE Int. Joint Conf. on Biometrics (IJCB), pages 1–6, Denver, CO, USA, 2017. IEEE. 1, 2

[22] D. Yambay, J. S. Doyle, K. W. Bowyer, A. Czajka, and S. Schuckers. Livdet-iris 2013 - iris liveness detection competition 2013. In IEEE Int. Joint Conf. on Biometrics (IJCB), pages 1–8, Clearwater, FL, USA, Sept 2014. IEEE. 2

[23] D. Yambay, B. Walczak, S. Schuckers, and A. Czajka. Livdet-iris 2015 - iris liveness detection competition 2015. In IEEE Int. Conf. on Identity, Security and Behavior Analysis (ISBA), pages 1–6, New Delhi, India, Feb 2017. IEEE. 2