A Biometric Security Model for Mobile Applications

SORIN SOVIANY[1], SORIN PUSCOCI[1], VIRGINIA SANDULESCU[1], CRISTINA SOVIANY[2]
Communication Terminals and Telematics (T.C.T.) [1]
National Communications Research Institute (I.N.S.C.C.) [1]
Bd. Preciziei, No. 6, Bucharest [1]
ROMANIA[1]
Features Analytics[2]
2, rue de Charleroi 1400 Nivelles[2]
BELGIUM[2]
sorin.soviany@inscc.ro[1], sorin.puscoci@inscc.ro[1], virginia.sandulescu@inscc.ro[1],
cristina.soviany@features-analytics.com[2], http://www.inscc.ro[1] https://features-analytics.com/[2]

Abstract: - A biometric security model for mobile applications is defined. It is a low-complexity design with a security architecture including 2 biometric traits (fingerprint and iris). The fingerprint processing for feature generation is optimized on the mobile device, but the iris template optimization is performed on server. In both cases, the feature space is transformed to provide a suitable trade-off performance vs. complexity for a properly reduced dimensionality. The matching is based on a target-vs.-non-target classification in order to meet the requirements of an identification process in which only a target identity must be recognized. The target identity belongs to the mobile device owner.

Key-Words: - security model, feature, data fusion, mobile application

1 Introduction

The extending usage of the mobile applications is enabled by the technological advances in hardware, software and mobile networking. The mobility became a key factor for such applications design, requiring optimizations according to several constraints for processing, storage and transfer rate.

A critical issue for the mobile applications is the security. The conventional data protection mechanisms are constrained in this case by the storage and processing limitations. The security issues for mobile applications are generated by the emerging of the new threats. The inappropriate usage of the mobile devices, the bugs within the new apps, the authentication issues, the client data sensitivity, the mobile communication networks vulnerabilities are a few reasons to develop innovative security solutions for mobile applications.

The authentication remains one of the most important security mechanisms. For the conventional applications the multi-factor authentication is a common approach. The multi-biometric solutions (with several biometric traits) are already applied on large scale. The problem is how to use these methodologies for the mobile use-cases while ensuring at least the same performances as for the desktop apps under the specific constraints.

In this paper, a biometric security model for mobile applications is defined, with design for low-complexity applications, using 2 biometric traits (fingerprint and iris). The feature space is transformed to provide a suitable trade-off performance vs. complexity for a reduced dimensionality. The matching is performed with a target-vs.-non-target classifier in order to meet the requirements of an identification process for a target identity belonging to the mobile device owner, supporting the secured access to an application service such as m-Banking or m-Health.

The remainder of the paper has the following structure: Section 2-the general design of the security architecture for mobile applications; Section 3–the biometric data processing and experimental results; Section 4- conclusions.

2 The Security Architecture design for Mobile Applications

2.1 Actual technical developments

As concerning the mobile devices with password-free security, there are already available smartphones including the biometric authentication. The biometric approach for the smartphone security started to be largely considered only since 2013.

Currently many smartphones integrate several biometrics. This allows the multimodal biometric solutions development for mobile applications. The initial usage of biometrics for smartphones was to unlock these devices for reaching the home screens [1]. Now the biometrics usage concerns more sophisticated applications, such as the authentication for m-Banking, m-Health and other Application-layer mobile services.

The following biometric methods are considered for mobile users [1]:

- *fingerprint scanning*, firstly introduced in 2013 by Apple into iPhone 5S to unlock the device and later for user's authentication to secure a mobile payment application (Apple Pay), and then by other smartphones manufacturers with their devices (Samsung Galaxy S8, LG G6, Huawei Mate 10;
- *facial recognition*, for Samsung Galaxy S8, a Google and Samsung smartphone (Galaxy Nexus), also an Apple product iPhone X -with an application option via Apple Pay, beside its unlocking method Face ID. While Face ID provides a proper security to authenticate Apple Pay operations, the face recognition provided on many Android-based smartphones was not so secure for applications requiring a high degree of data protection. Recently, the facial recognition started to be considered again as a feasible option for mobile security applications, due to the technological advances in hardware, algorithms and their software implementation. Other mobile devices with facial recognition are Galaxy Note 8, LG V30, Huawei P10;
- *iris recognition*. Among the first smartphones with this capability one can mention Fujitsu NX F-04G and Microsoft Lumia 950 (2015); Samsung introduced this biometric in 2016 (Galaxy Note 7) and then in 2017 for Galaxy S8 and Galaxy Note 7. So far, the iris recognition on Android smartphones is not considered as secure enough for the authentication in m-Banking, despite of the high performance provided by iris recognition for desktop applications. The main usages remained the phone unlocking;
- *voice recognition* is included in LG V30 with a voice-unlock capability (Voice Print). The Android operating system integrates voice recognition through the Smart Lock setting.

As concerning the applications with biometric credentials for mobile devices, one can mention some recent developments.

A secured Android application including biometric authentication is presented in [2]. The security mechanism includes cryptographic operations and fingerprint samples registration on the mobile device. The main usage in this case is for the device unlocking.

A review of facial recognition actual applications for smart devices (with focus on mobile ones) is given in [3]. It explains why the facial recognition could have a reliable potential to become a trusted form of authentication, even for mobile use-cases. The ensured security degree is explored considering the use-cases of mobile device applications available for Android and iOS platforms. iOS facial recognition apps are more secure than the Google Play store apps for Android devices.

A biometric authentication method for Android-based mobile devices is presented in [4]: a fingerprint-based authentication system with implementation on a LG Nexus 5 device.

The recognition of fingerprints on mobile applications is approached in [5] with an Android case study. This is an application with fingerprint recognition using camera from the mobile device.

The facial identification on Android smartphones is approached in [6]. An implementation of biometric identification for large datasets is presented, together with its performance evaluation.

Another use-case is a biometric authentication through a virtual keyboard for smartphones [7]. The focus is on the authentication with keystroke dynamics, with new features that are extracted for this biometric when applied for mobile devices.

These are only a few examples of recent developments of biometric authentication solutions for mobile devices looking to secure applications with high- and medium-sensitivity data.

The integration of several biometrics (fingerprint, face and/or iris, voice recognition as an additional trait) on the same mobile device could represent a useful option, even for the Android devices; several biometrics provide a higher security, but with a careful consideration for the complexity issues. The potential of multimodal approaches was proved in many security applications, such as the example of a cryptographic key generation with multiple biometric modalities given in [8].

2.2 The security architecture specification

The security architecture specification includes the overall security model (section 2.2.1) with its functional components (section 2.2.2). The data processing for the mobile user authentication is detailed within the section 3.

2.2.1 The overall security architectural model

The overall security model is defined for the target mobile application architecture depicted in fig.1.

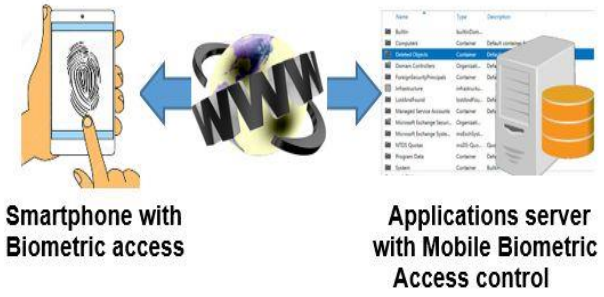

Fig.1: The application general architecture

This general architecture is typical for use-cases such as m-Health and m-Banking, involving the remote access to large databases storing sensitive information. The remote applications are optimized for mobile users with their smartphones, in order to meet the constraints of the mobile devices. For this architecture a security model is defined to provide the mobile user authentication with fingerprint and iris. The security model is depicted in fig. 2.

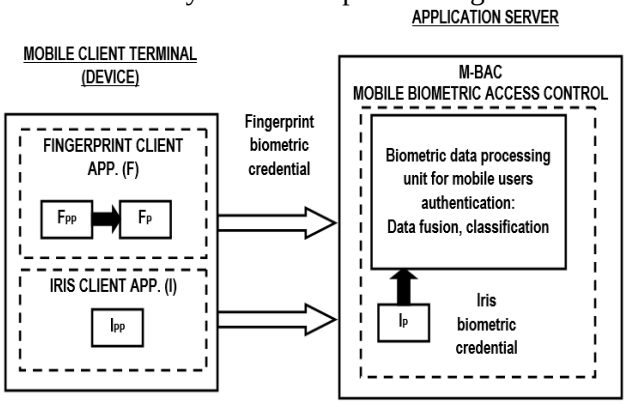

Fig. 2: The security model

The security architecture is based on the typical client-server model, but with distributed biometric samples processing capabilities among the mobile client device and the application server.

2.2.2 The basic functional components

The security model (fig. 2) shows the functional modules for the mobile client and server.

The biometric data processing capabilities are distributed between the mobile client device and the remote application server (with M-BAC, Mobile Biometric Access Control):

- *client*: a Fingerprint Client App.(F) and an Iris Client App. (I). The Fingerprint module includes 2 functions: F_{PP} (Fingerprint Pre-Processing) and F_P (Fingerprint Processing). The Iris module includes only one function: I_{PP} (Iris Pre-Processing). The advanced iris processing (I_P module) is performed on server. For each biometric, the PP (Pre-Processing) function performs the samples processing for feature extraction to generate an initial feature set: $FV_{0,F}$ (fingerprint feature vector) and $FV_{0,I}$ (iris feature vector). The P (Processing) function performs more advanced feature space transformations and feature selection to provide the best features with the suitable discriminant power. F_P generates the fingerprint credential that has to be sent to the authentication module on the server (M-BAC). The iris credential is generated based on the initial feature vector $FV_{0,I}$ but on the server;
- *server*: the iris processing module I_P, that generates the iris biometric credential using the received initial feature set. The M-BAC (Mobile-Biometric Access Control) module performs: data classification for each biometric, data fusion and the authentication.

3 The Data Processing for the Mobile Users Authentication

The overall data processing is performed according to fig. 3. This includes ***feature generation***, ***data fusion*** and ***matching***.

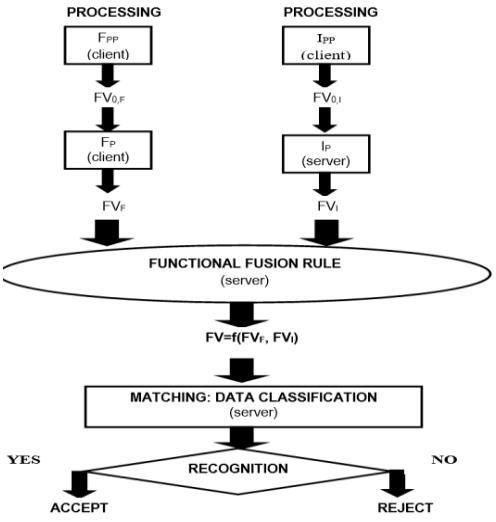

Fig. 3: The overall biometric data processing

The following major operations are performed:

- *the fingerprint credential generation*: pre-processing (F_{PP}) and processing (F_P), both on client. This is the ***feature generation*** for the 1st biometric (fingerprint). The resulting credential FV_F is sent to the server for the next operations (data fusion, matching);

- *the iris credential generation*: pre-processing (I_{PP}) on client and processing (I_P) on server. The iris credential FV_I is generated on server and then used within the next operations (data fusion, matching);
- *the data fusion*: the combination process between the 2 biometric credentials, providing a single feature vector to the matching stage;
- *the matching (data classification)*: the advanced stage of data processing for the identity validation.

This ongoing research exploits our previous works about the feature fusion [9],[10],[11].

3.1 Feature generation

In the ***feature generation*** the raw samples are processed to perform: *feature extraction*, with an image processing algorithm; *feature space transformation*, improving the discriminant power of the features; *feature selection*, optimizing the dimensionality and preserving the informative features.

3.1.1 Feature extraction

For the both biometrics the feature extraction is performed using the same algorithm to simplify the design, given the constraints of the mobile devices. Another reason is to ensure the homogeneity of the feature vectors. The homogeneity is the condition for the functional-based feature fusion feasibility, requiring the same dimensionality of the vectors.

The feature extraction for both biometrics is depicted in fig. 4 (F_{PP}, I_{PP} modules). A textural and regional approach with 2nd order statistical features is applied. These features are computed using Co-occurrence matrices (COM) from the regions of interest (ROIs) that are selected within the original images. The ROI definition and selection are manually done, instead of the automatic approaches as in most of the existing developments. The input images are captured using the mobile devices (smartphones) camera and no dedicated devices, which is important especially for the iris recognition, where a lot of feature points are typically involved. We are exploring the effects of different parameterizations on the recognition performance.

This enables the adjustment of the resulted dimensionality through a proper setting of the COM-based Feature Extractor parameters. The Co-occurrence matrix must contain less null values and more significant values.

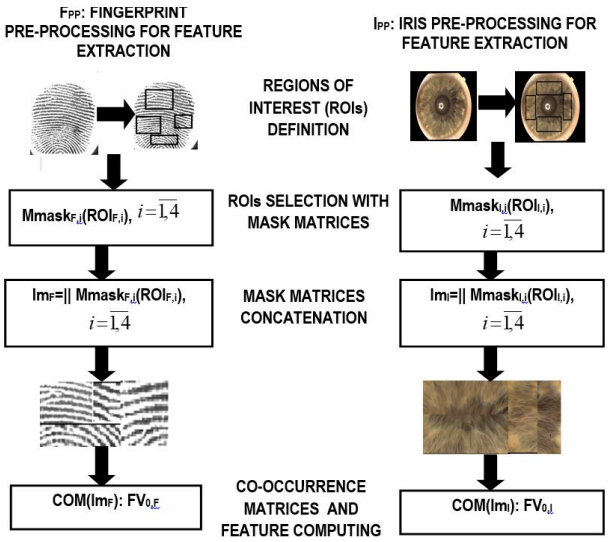

Fig. 4: The Feature Extraction (modules F_{PP}, I_{PP})

The *Feature Space Size* (*FSS*) adjustment allows the curse of dimensionality and classification performance peaking prevention [12]. In this way, the design provides a reliable trade-off complexity vs. performance, also ensuring the feature vectors homogeneity to avoid the concatenation-based feature-level fusion.

Before the feature extraction, the original images are converted to gray-scale using the method presented in [13], which is appropriate for the textural analysis.

The following operations are performed for the both biometrics (according to fig. 4):

- *ROI definition*, within the following sub-stages:
  - ✓ setting the number of ROIs to be extracted from the input image, the same amount for both biometrics: $n_{ROI,F} = n_{ROI,I} = 4$;
  - ✓ fixing the rectangular areas that contain the meaningful details from the original images. This is done by specifying the initial coordinates ($x_{B,i}$, $y_{B,i}$) and their offsets ($\Delta x_{B,i}$, $\Delta y_{B,i}$). The ROI specification is as following:

    $$ROI_{B,i} : \left(x_{B,i}, x_{B,i} + \Delta x_{B,i}, y_{B,i}, y_{B,i} + \Delta y_{B,i} \right) \quad (1)$$

    where $i = \overline{1,4}$ and $B \in \{F, I\}$ (F fingerprint, I iris). The ROIs could be specified also with the upper-left and lower-right points;
- *ROI selection*, using mask matrices (*Mmask*) to extract the previously defined ROIs. The mask matrices are defined like in [14]:

$$Mmask(ROI_{B,i})[i_{B,i}, j_{B,i}] = \begin{cases} 1, x_{B,i} \le i_{B,i} \le x_{B,i} + \Delta x_{B,i}, \\ y_{B,i} \le j_{B,i} \le y_{B,i} + \Delta y_{B,i} \\ 0, otherwise \end{cases} \quad (2);$$

- *Mask matrices concatenation*, to generate a single image by the previously extracted ROIs

fusing. The concatenation is done horizontally and vertically, with care about the sizes (the same number of rows for the horizontal concatenation, the same number of columns for the vertical concatenation). The output image is

$$\mathrm{Im}_B = \|_{i=1}^{4} Mmask(ROI_{B,i}) \quad (3)$$

- *Statistical features computing using Co-occurrence matrices (COM)*. The next step is to compute the Co-occurrence matrices from the previously extracted ROIs for each biometric. The resulting 2nd order statistical features evaluate the gray levels distribution within the images [12]. Each COM element estimates the probability of a certain gray level for one pixel within the image, while another pixel with a given displacement has another gray level. The COM definition, based on the statements given in [15],[16], is:

$$COM_{\Delta x_B, \Delta y_B}(\mathrm{Im}_B)\lfloor u_B, v_B \rfloor = P\{\mathrm{Im}_B(x_B, y_B) = u_B, \mathrm{Im}_B(x_B + \Delta x_B, y_B + \Delta y_B) = v_B\} \quad (4)$$

where: $B \in \{F, I\}$, Δx_B and Δy_B are the horizontal and vertical displacements of the pixels. This approach exploits the textural properties of the image [15],[16]. The resulting feature space dimensionality (*FSS*) is adjusted by varying the feature extractor parameters to achieve Co-occurrence matrices with many relevant (non-null) values. The following amounts are the parameters of the feature extractor [9],[16]:

  - N_{GLB} (number of Gray-Level Bins), that provides the most informative features by increasing the number of the significant values within the resulted COM and minimizing the number of null values. The settings are the following: 1) $N1_{GLB,F}=6$, $N1_{GLB,I}=8$; 2) $N2_{GLB,F}=4$, $N2_{GLB,I}=6$. This allows to modify the dimensionality (feature space size);
  - *OFFS* (offset), a measure of the displacement, in number of pixels, between the pixels pairs that are used to compute COM. It must not exceed a certain value in order to not increase too much the resulting spacing, therefore reducing the overall number of pixel pairs. The setting is: $OFFS_B=2$.

  The resulting feature vectors are $FV_{0,F}$ (fingerprint) and $FV_{0,I}$ (iris). Their dimensionality is given by

$$FSS_{0,B} = size(FV_{0,B}) = \left(N_{GLB,B}\right)^2 \quad (5)$$

where $B \in \{F, I\}$ (F fingerprint, I iris).

3.1.2 Feature space transformation and feature selection

The feature space transformation and feature selection optimize the dimensionality preserving the most informative features, to meet the low-complexity requirements of the mobile applications. The dimensionality adjustment is depicted in fig. 5. The corresponding operations are performed by the modules F_P and I_P.

The following operations are performed:

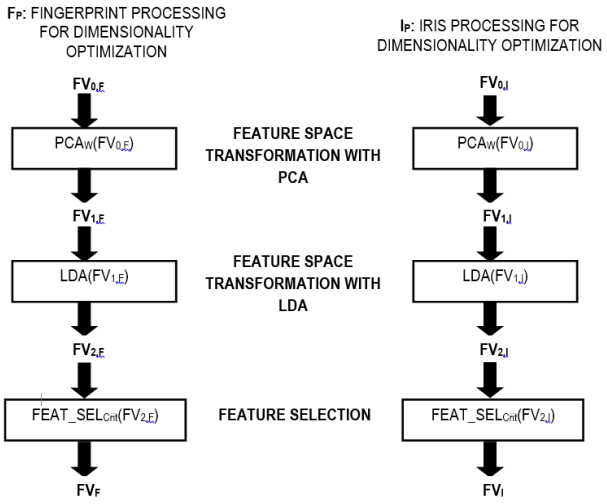

Fig. 5: The Feature Space Optimization (modules F_P, I_P)

- *Feature space transformation with PCA (Principal Component Analysis)*: to maximize the overall variance but reducing the features correlation [12],[17]. PCA retains the most informative dimensions. The typical PCA algorithm is an unsupervised data projection and it does not always preserve the class separation. Here a supervised version is used with a covariance matrix *CM* that is weighted with the class priors, according to:

$$CM_B = \sum_{j=1}^{C} p_j \cdot CM_{B,j}, B \in \{F, I\} \quad (6)$$

where: CM_B is the weighted covariance matrix for the biometric dataset B (fingerprint or iris); $CM_{B,j}$ is the covariance matrix for the samples belonging to class j; p_j is the prior for the class j within the training dataset; C is the number of classes ($C=2$). The resulting vectors are $FV_{1,F}$ and $FV_{1,I}$;

- *Feature space transformation with LDA (Linear Discriminant Analysis)*: to maximize the class separation. LDA is a linear transformation w that maximizes the Fisher criterion (FDR, Fisher Discriminant Ratio) -the ratio between the inter-class and intra-class variance [9],[12]:

$$\mathrm{FDR}(w) = \frac{\sigma^2_{\mathrm{inter-class}}}{\sigma^2_{\mathrm{intra-class}}} = \frac{w^T \cdot S_{Between} \cdot w}{w^T \cdot S_{Within} \cdot w} \quad (7)$$

$S_{Between}$ is the between-class scatter matrix and S_{Within} is the within-class scatter matrix [17]. The resulting vectors are $FV_{2,F}$ and $FV_{2,I}$;

- *Feature selection*: to reduce the dimensionality but preserving the relevance and enhancing the discriminant power. The goal is to achieve the same feature space size (*FSS*) for both biometrics. The most informative features are retained using a non-exhaustive feature selection method with a performance criterion: $\text{FEAT\_SEL}_{\text{Crit}}(\text{FV}_{2,\text{B}})$ (fig. 5). Several options are considered for the *feature selection algorithm* [12]: forward-searching, backward-searching, floating-searching, individual ranking and random selection. The individual ranking provides the best execution time for the available datasets if *FSS* does not exceed 70 (which is our case). The *performance criterion* is the 1-NN (Nearest Neighbor) rule, because it provides the limitation of the classification error rate, according to [18]:

$$\varepsilon^* \le \varepsilon_{1-NN} \le 2\cdot\varepsilon^*\cdot\left(1-\varepsilon^*\right) \le 2\cdot\varepsilon^* \quad (8)$$

where ε^* is the error rate for the optimal Bayes classifier and ε_{1-NN} is the error rate for the 1-NN classifier. The resulting feature vectors are FV_F and FV_I, with the same size: $FSS_B = size(FV_B) = 12, \text{B} \in \{F, I\}$.

3.2 Data Fusion and Matching

The last operations are ***data fusion*** and ***matching*** (***data classification***) for the recognition (fig. 3).

3.2.1 Data Fusion

The ***data fusion*** combines data from independent biometric sources of the same person to generate a global decision, score or feature set for the recognition application. The fusion could be performed at several processing stages, *pre-* or *post-classification* [19]. The *post-classification fusion* is the most implemented one due to its simplicity, but with the cost of some loss of information.

In this research the target is the *pre-classification feature-level fusion*. This is still a challenge for the biometric solutions design, given the variety of the feature extraction algorithms (with incompatibilities among the feature sets) and the difficulty to find relationships among the different feature spaces [19].

The common approach for **feature-level fusion** is the ***concatenation***, as it does not require **homogeneous** vectors. This is still expensive in terms of dimensionality, leading to the curse of dimensionality. Our goal is to avoid the ***concatenation-based fusion*** and to define a ***functional-based feature-level fusion***, preventing the dimensionality increasing. The feasibility of the ***functional fusion*** is provided by the feature vectors **homogeneity** (the **common dimensionality** of the feature spaces, FSS_B).

The ***functional feature fusion*** model is $FV = f(FV_F, FV_I)$; f is the function that uses the fingerprint and iris feature vectors to compute the fused feature vector *FV*. The following ***functional fusion*** rules are considered, together with a weighting (W_F for fingerprint, W_I for iris):

- *R1: the weighted average rule*, given by

$$\text{FV[k]} = \frac{W_F\cdot FV_F[k] + W_I\cdot FV_I[k]}{W_F + W_I},\ \text{k} = \overline{0, FSS_B - 1} \quad (9);$$

- *R2: the weighted sum rule*, with 2 variants:

a) $\text{FV[k]} = W_F\cdot FV_F[k] + W_I\cdot FV_I[k],\ \text{k} = \overline{0, FSS_B - 1}$ (10)

b) $\text{FV[k]} = W_F\cdot FV_F[k] + W_I\cdot FV_I[\text{FSS}_B - k]$ (11)

- *R3: the product rule*, with 2 variants:

a) $\text{FV[k]} = FV_F[k]\cdot FV_I[k], \text{k} = \overline{0, FSS_B - 1}$ (12)

b) $\text{FV[k]} = FV_F[k]\cdot FV_I[FSS_B - k]$ (13)

3.2.2 Data Classification

A SVM (Support Vector Machine) classifier is used, as in [11], but with differences concerning the kernel and the classes.

This classifier is suitable for the available data, showing a good stability. The application requires a target identification in which the most important identity must be recognized (as a target class C1), while all the other identities are included into the 2nd class (non-target) C2. The target identity belongs to the mobile device owner.

The *Training Set Size* (*TSS*) is fixed using the condition given in [12], $2 \le \frac{TSS_B}{FSS_B} \le 10, B \in \{F, I\}$, that provides an optimal range for the peaking and curse of dimensionality prevention [12]. Given FSS_B, the optimal TSS_B should be between 24 and 120 samples per class. The original dataset with 80 samples per class is divided into 2 independent subsets, for training (60 samples) and for testing (20 samples).

The kernel SVM model is given by [12],[17]:

$$g(x_{test}) = \text{sgn}\left(\sum_{tr=1}^{TSS_B}\alpha_{tr}\cdot y^{tr}\cdot K\left(x_{tr}, x_{test}\right) + w_0\right) \quad (14)$$

in which [11]: x_{test} is the current testing sample; x_{tr} is the training sample; α_{tr} is the Lagrange multiplier that is used to find the maximum margin hyper-

plane; w_0 is the offset parameter; y^{tr} is the class label ($y^{tr}=1$ for the target class C1, $y^{tr}=-1$ for the non-target class C2); $K(.,.)$ is the kernel that enhances the classifier behaviour by ensuring a transformed space with a higher linearity. A generalized polynomial kernel is applied:

$$K(x_{tr}, x_{test}) = (a \cdot x_{tr} \cdot x_{test} + b)^p, p \in \{1,2\} \quad (15)$$

The coefficients a and b are fixed using the experimental data.

3.3 Experimental achievements

We used a dataset with images from 40 persons, images that were taken within our own research. The overall dataset contains 4 images per individual, from which the best quality image is selected.

The performance is evaluated for each of the 5 feature-level fusion rules (the weighted average, the weighted sum with 2 variants, the product with 2 variants) and for the 2 polynomial kernels (1st and 2nd degree). The measures are TPR (True Positive Rate for the target identity class) and FPR (False Positive Rate), using the ROC (Receiver Operating Characteristic) curves representation for several thresholds. The curves are shown in fig. 6(a, b).

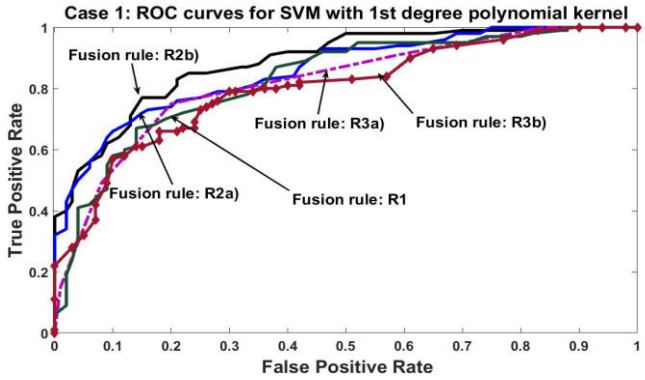

Figure 6a: The performances for the 5 fusion rules and polynomial kernel: 1st degree

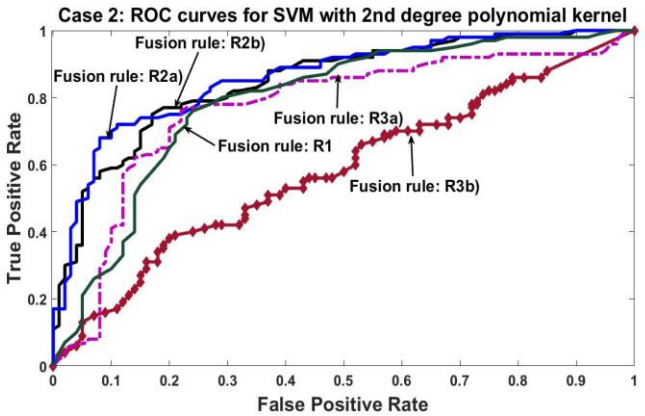

Figure 6b: The performances for the 5 fusion rules and polynomial kernel: 2st degree

The goal is to find the best feature fusion rule for a certain $\frac{TSS_B}{FSS_B}$ ratio, for the both cases of the kernel SVM. The best feature fusion rule should provide the optimal $\frac{TPR}{FPR}$ ratio, as resulting from the ROC analysis.

From fig. 6(a, b) one can see that the best performances are achieved in the 1st case (polynomial kernel 1st degree) with the fusion rule R2b) (the 2nd version of the weighted sum rule), ensuring a TPR close to 80% for a FPR close to 15%. These results are obtained for a reduced fused feature space with only 12 features, a size that is significantly lower that other approaches with feature-level fusion (especially concatenation). On the other hand, the 2nd degree polynomial kernel SVM seems to have a lower performance, according to fig. 6b); the best operating point does not have the same TPR vs. FRP ratio as in the 1st case.

There is still a good potential for the further improvements, having as the target a TPR around 90% for a FPR around 5%. The further improvements could be achieved by working on the classifiers parameters and hyper-parameters (the training set sizes), also with a further adjustment of the feature space, for instance by generating some new features starting from what are already extracted and by additionally transforming the feature space in order to increase the discriminant power. New functional fusion rules could be defined to enhance the recognition performances.

4 Conclusion

A security architecture for smartphone-based applications is defined, in order to evaluate the reliability of the mobile users' biometric authentication for high-sensitivity data applications. The model uses 2 biometrics, fingerprint and iris, with several feature fusion rules. The fusion is based on the functional combinations of the input vectors, avoiding the concatenation and the curse of dimensionality.

The particularities of this security model for mobile applications result from the *data acquisition* and *processing*. For the *data acquisition* only the smartphone camera are used, no dedicated biometric devices. This is a reason for a careful feature selection to ensure an optimal discriminant power. The *data processing* supports a low-complexity design especially as concerning the feature generation (the same feature extraction algorithm for both biometrics). We exploited some of our

previous works, but with differences as concerning: the number of the selected ROIs; the images sources; the parameterization of the feature extractor; testing of several functional feature fusion rules; using a polynomial kernel SVM model with several degrees. A further step is the software implementation with evaluations for the execution time vs. the security performances.

References:

[1] F. Agomuoh, Smartphone biometrics are no longer the stuff of science fiction, *Business Insider*, Dec. 27, 2017,

[2] Paulson P K., Ambili K, Secured Android Application Using Biometric Authentication, *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 5, Issue 4, April 2017, pp. 7715-7719

[3] M. G. Galterio, S. A. Shavit and T. Hayajneh, A Review of Facial Biometrics Security for Smart Devices, *Computers* 2018

[4] V. Conti, M. Collotta, G. Pau, and S. Vitabile, Usability Analysis of a Novel Biometric Authentication Approach for Android-Based Mobile Devices, *Journal of Telecommunications and Information Technology*, No. 4, 2014

[5] O. Dospinescu and I. Lîsîi, The Recognition of Fingerprints on Mobile Applications – an Android Case Study, *Journal of Eastern Europe Research in Business and Economics*, Vol. 2016 (2016)

[6] S. F. Darwaish, E. Moradian,T. Rahmani, M. Knauer, Biometric identification on android smartphones, *18th International Conference on Knowledge Based and Intelligent Information & Engineering Systems - KES2014*, Procedia Computer Science 35 (2014), pp. 832-841

[7] M. Trojahnand, F. Ortmeier, BIOMETRIC AUTHENTICATION THROUGH A VIRTUAL KEYBOARD FOR SMARTPHONES, *International Journal of Computer Science & Information Technology (IJCSIT)* Vol 4, No 5, October 2012

[8] A. Jagadeesan, T.Thillaikkarasi, K.Duraiswamy, Cryptographic Key Generation from Multiple Biometric Modalities: Fusing Minutiae with Iris Feature, *International Journal of Computer Applications (0975 – 8887)* Volume 2 – No.6, June 2010

[9] S. Soviany, C. Soviany, S. Puşcoci, A Multimodal Biometric System with Several Degrees of Feature Fusion for Target Identities Recognition, *The 2016 International Conference on Security and Management (SAM'16)*, Las Vegas, USA, July 25 – 28, 2016.

[10] S. Soviany, V. Săndulescu, S. Puşcoci, C. Soviany, M. Jurian, An Optimized Biometric System with Intra- and Inter-Modal Feature-level Fusion, *ECAI 2017 - International Conference – 9th Edition Electronics, Computers and Artificial Intelligence*, România, 29 June -1 July, 2017.

[11] S. Soviany, V. Săndulescu, S. Puşcoci, C. Soviany, A Biometric System with Hierarchical Feature-level Fusion, *ECAI 2018 - International Conference – 10th Edition Electronics, Computers and Artificial Intelligence*, România, 28-30 June, 2018

[12] S.Theodoridis, K. Koutroumbas: Pattern Recognition, 4th edition, A*cademic Press Elsevier*, 2009

[13] D.Bhattacharyya, P. Das, S.K.Bandyopadhyay, T.Kim: IRIS Texture Analysis and Feature Extraction for Biometric Pattern Recognition, *International Journal of Database Theory and Application*, vol. 1, nr. 1, pp. 53-60, December 2008

[14] S. Soviany, C. Soviany: A Biometric Security Model with Identities Detection and Local Feature-level Fusion, *The 2013 International Conference on Security and Management (SAM'13)*, World Academy of Science, Las Vegas, SUA, July 22-25, 2013

[15] A.Eleyan, H.Demirel: Co-occurrence matrix and its statistical features as a new approach for face recognition, *Turk J Elec Eng & Comp Sci*, Vol.19, Nr.1, 2011

[16] S. V. Bino, A. Unnikrishnan, B.Kannan: Gray level Co-Occurrence Matrices: Generalisation and some new features, *International Journal of Computer Science, Engineering and Information Technology* (IJCSEIT), Vol.2, No.2, April 2012

[17] D. Zhang, F. Song, Y. Xu, Z. Liang: Advanced Pattern Recognition Technologies with Applications to Biometrics, *Medical Information Science Reference*, IGI Global, 2009

[18] L.Devroye, L.Gyorfy, G.Lugosi: A Probabilistic Theory of Pattern Recognition, *Springer*, 1997

[19] A. Jain A., K. Nandakumar K., A. Ross: Score Normalization in multimodal biometric systems, *Pattern Recognition, The Journal of the Pattern Recognition Society*, 38 (2005)