

Application-Specific Biometric Templates

James L. Cambier, Ph.D.
VP Research
Iridian Technologies, Inc.
Moorestown NJ
jcambier@iridiantech.com



Co-Authors

Michael Braithwaite
Ulf Cahn von Seelen
James Cambier
John Daugman
Randy Glass
Russ Moore
Ian Scott



Problem Statement

- Biometric technologies capable of identification enable the establishment of centralized authentication servers, in which a single database of enrolled templates provides authentication services for a wide variety of applications
- The potential for shared access and multiple uses of biometric databases raises concerns with respect to personal privacy
- If an enrolled biometric template is compromised it cannot be reissued like a password – it is gone forever



Comparison With Other Techniques

- Ratha et al ("Cancelable Biometrics", BC2000) describe applying irreversible transformations to raw biometric data – solves problem of re-usability but does not permit controlled sharing of templates
- Proposed technique is NOT designed to replace encryption – it is a form of weak (at best) symmetric encryption



Potential Solutions

- Prohibit compilation of identification databases
 - Negates convenience and other advantages of identification biometrics
 - Requires storage of reference templates on smart cards or other tokens that can be lost, destroyed, compromised, etc.
- Encrypt stored templates
 - Requires decryption before matching, exposing templates to hackers during match process
 - Increases processing requirements for matching
 - Requires existence of public key infrastructure



Application-Specific Biometric Templates

- Transform (or create) each template so that it assumes a unique format for each application
- Provide a controlled means for converting templates from one format to another so that they can be shared among applications subject to user authorization, eliminating the need for re-enrollment for every application
- Design transformations so that matching can be performed on transformed templates



Matching Functions

Consider biometric templates T_1 and T_2 derived from the same biologic entity (hand, finger, eye, etc.) such that an appropriate matching function $M(T_1, T_2)$ has a value

$$M(T_1, T_2) = 1$$

if the templates are judged to match (i.e. they came from the same biologic entity) and

$$M(T_1, T_2) = 0$$

if the templates are judged to not match.



Transformation Properties

A transformation F_A may be applied to the “root” templates T_1 and T_2 so that the transformed templates $F_A(T_1)$ and $F_A(T_2)$ have a unique format specific to a particular use or application A.

The transformation F_A should have the property that the matching process is invariant under the transformation, that is,

$$M(F_A(T_1), F_A(T_2)) = M(T_1, T_2)$$



Transformation Properties

If we define two different transformations F_A and F_B for applications A and B then

$$M(F_A(T_1), F_B(T_2)) = 0$$

even if T_1, T_2 are from the same biological entity, while in that case

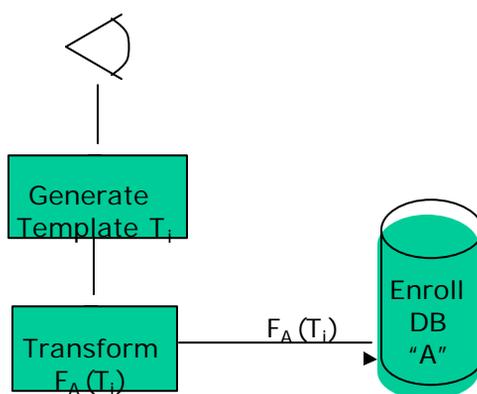
$$M(F_A(T_1), F_A(T_2)) = 1 \text{ and}$$

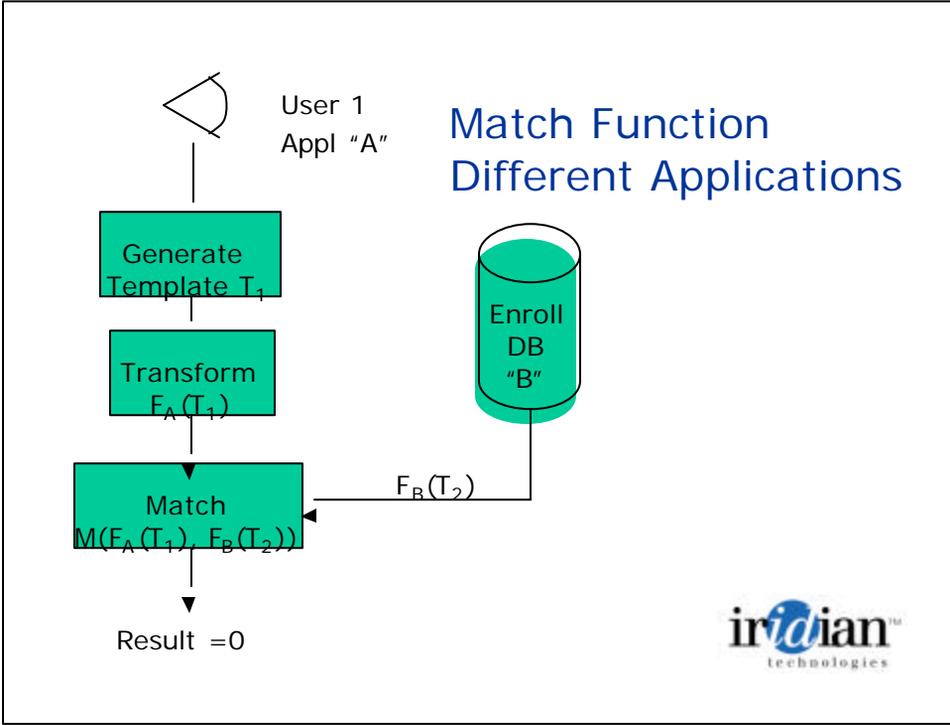
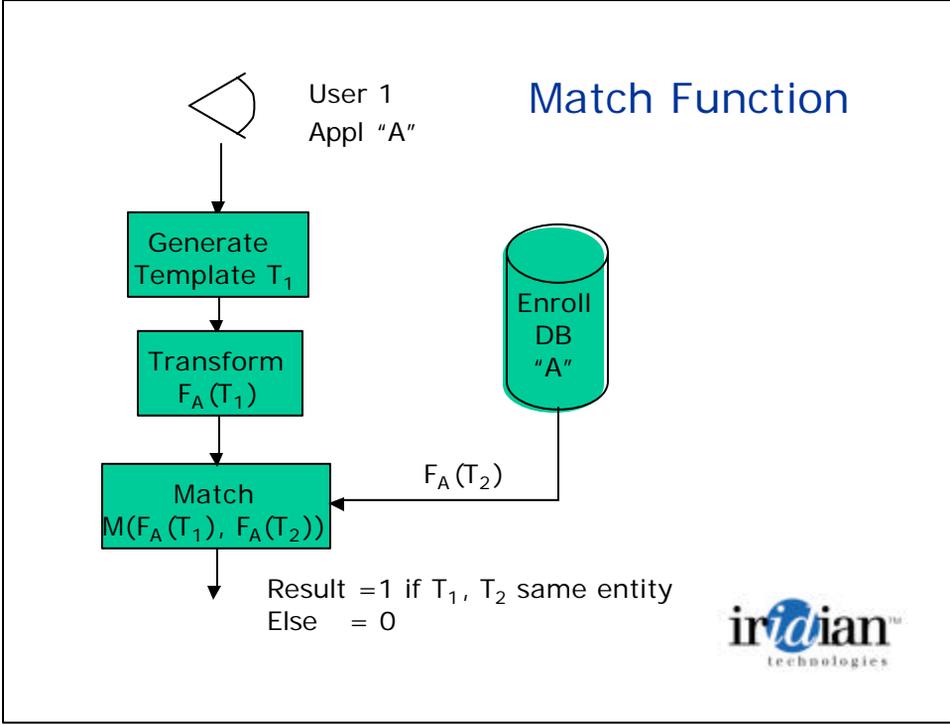
$$M(F_B(T_1), F_B(T_2)) = 1$$

This property assures that a template generated for one application A cannot be used for another application B.



Enrollment Process





Transformation Properties

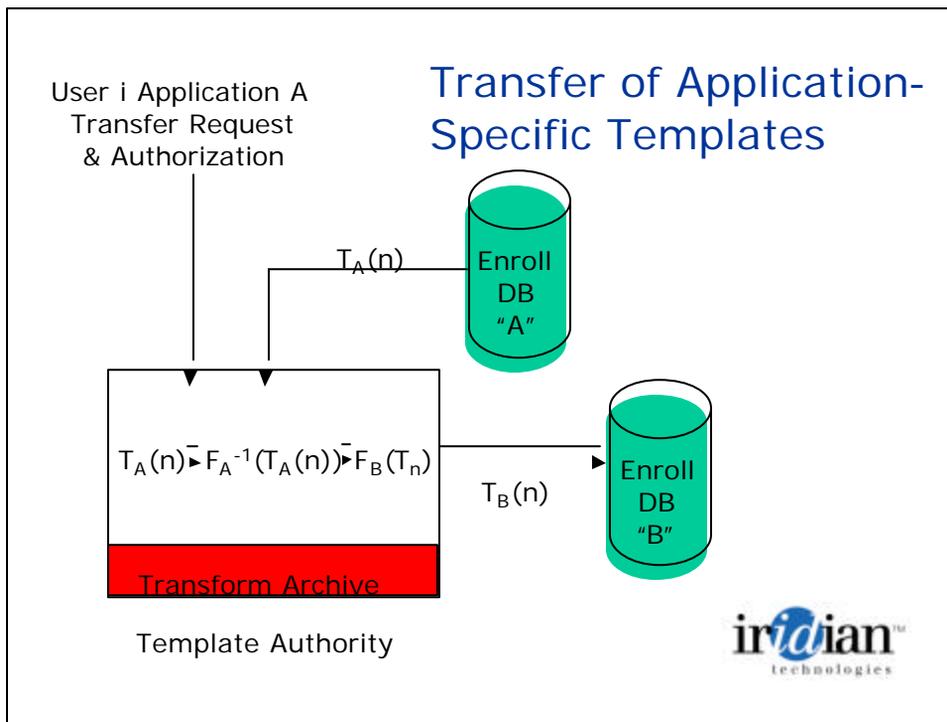
Transformations can be processed to create new templates. So if we have template $F_A(T_1)$ we can define transformation $F_{A,B}$ such that

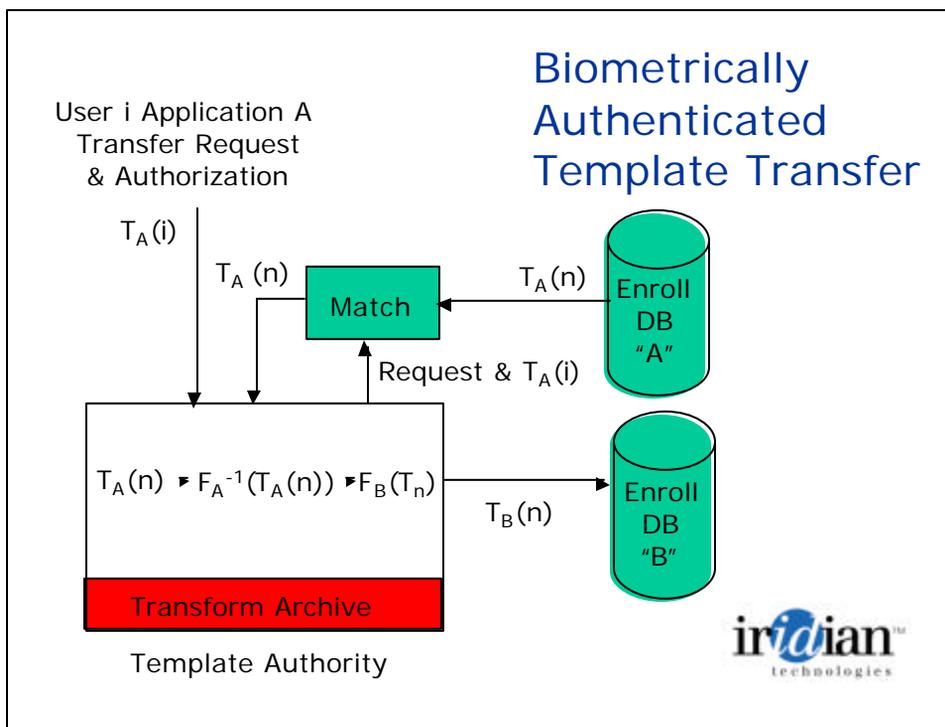
$$F_B(T_1) = F_{A,B}(F_A(T_1))$$

$$\text{i.e. } F_{A,B} = F_B F_A^{-1}$$

where F_B is the format created for Application B.

A user can authorize the custodian of database A to make his or her enrolled template available to the Application B database after application of transformation $F_{A,B}$ to change its format.





- ### Database Reissue
- Database custodian suspects or determines that its biometric data has been compromised, or its format has been discovered
 - Template Authority is asked to define a new transformation for its entire database, changing its format and rendering the stolen templates completely useless
 - Functionally equivalent to changing the password for a computer if it is determined that the password has been stolen.
- iridian™**
technologies

Client-Server Applications

- Previously enrolled user for application “A” wishing to be authenticated requests from the server a unique transformation “seed” number or key, from which a transformation can be generated.
- Server generates a random seed denoted “X”, transmits the seed X to the client, and at the same time computes the transformation
$$F_{X,A} = F_A F_X^{-1}$$
and saves it in temporary storage.
- Server then deletes X, F_X , and F_X^{-1} .



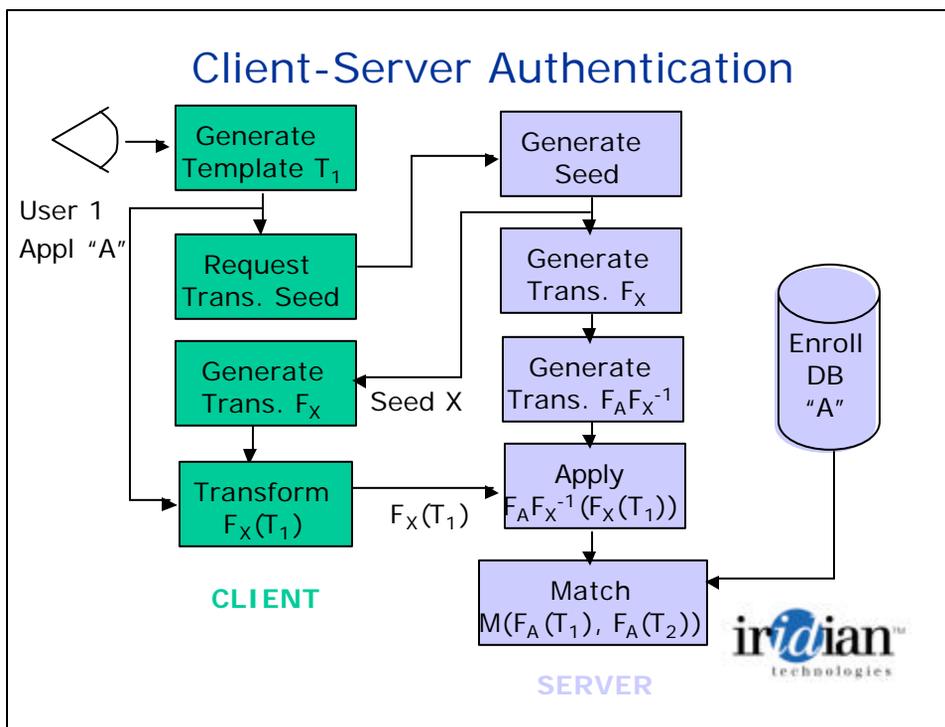
Client-Server Applications

- Client uses X to generate its own copy of F_X
- Client captures an image and generates a biometric template using F_X to transform the root template T_1 to the format prescribed by X.
- Server uses its temporarily stored transformation $F_{X,A}$ to convert the client's template to a format compatible with database A:

$$\begin{aligned} F_A(T_1) &= F_{X,A} (F_X(T_1)) \\ &= F_A F_X^{-1}(F_X(T_1)) \end{aligned}$$

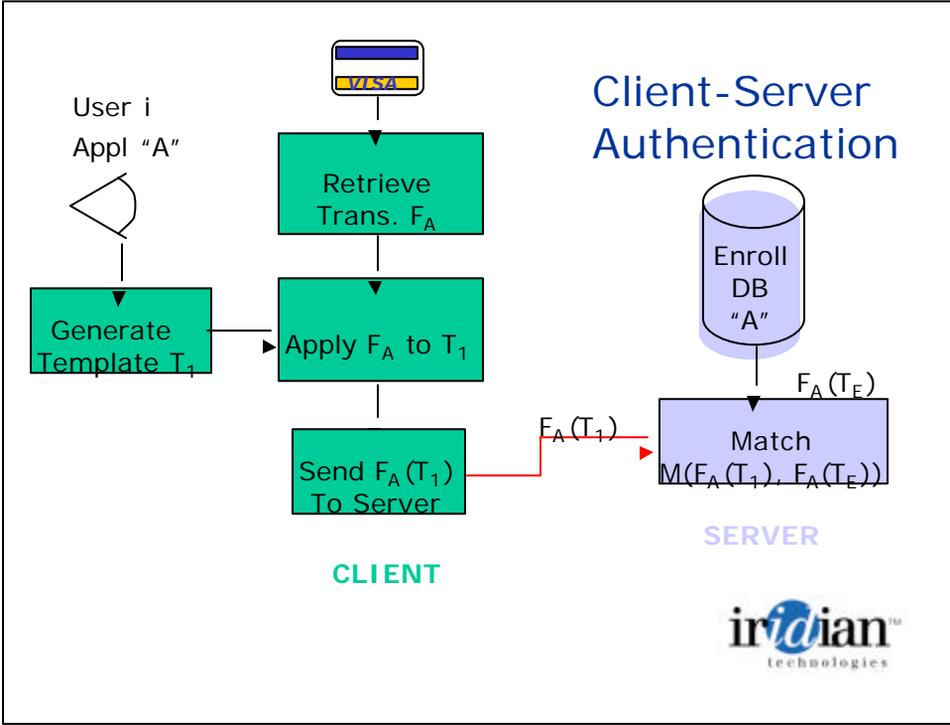
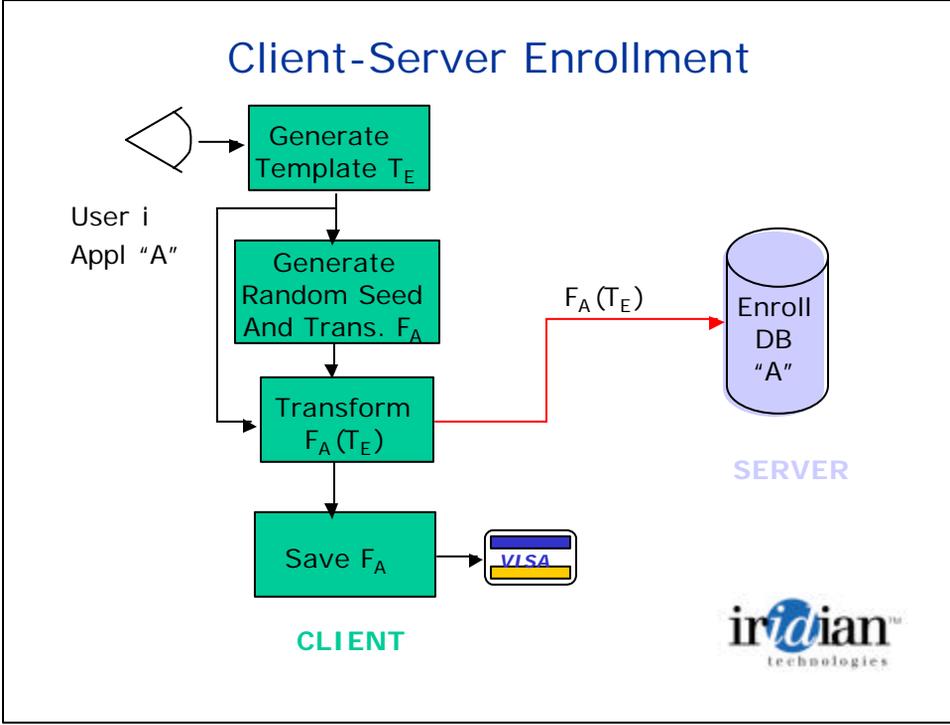
- Client's template has been generated and transmitted to the server in a unique format valid for only this single transaction





Client-Server Applications

- Before enrollment is performed, client application generates a random seed number and computes its own unique "A" transformation
- This transformation is applied to the enrollment template before sending it to the server.
- Transformation is also stored on a smart card or other portable media that the user keeps in his possession.
- The user may perform enrollments (with different transformations) for a number of applications, each time saving the appropriate transformation in portable storage.
- Each template in the enrolled database will have its own unique format, known only to its user.



Template Properties

- Biometric templates must be composed of an array $[t_1, t_2, t_3, \dots, t_n]$ of independent data entities t_i , which may be isolated binary bits or groups of bits
- Matching function is one that judges the similarity between two templates by examining corresponding independent data entities, such as the Hamming Distance $HD(T_1, T_2)$ which examines every pair of corresponding bits in templates T_1 and T_2 and counts the proportion of bits that differ between the two templates
- HD concept can be generalized to larger data entities, counting the number of corresponding entities that are not identical



Transformation Properties

- A suitable transformation F used for such biometric templates must have three properties:
 - F must not alter the length of the template
 - F must not change the value of the control bits, if used
 - F must not alter the number of matching (or mismatching) data bit pairs
- One such transformation is simple permutation, which simply alters the position of some or all data bits (or bytes)
 - n independent entities implies $n!$ possible transformations
 - $256! = 8.6 \times 10^{506}$ (using Stirling's approx.)
 - $2048! = 10^{5894}$



Transformation Properties

- Another form of transformation is based on the logical exclusive-or (XOR) function.
- Single bit values are XORed with a predefined mask function. If T_i is the i^{th} data bit of template T and M_i is the i^{th} mask bit then the i^{th} transformed template bit is

$$F_i(T) = T_i \text{ XOR } M_i$$

- XOR function changes the value of any bit for which the corresponding mask bit is a 1. If the template has 2048 data bits the number of possible masks is $2^{2048} = 3.2 \times 10^{616}$
- Mask should contain 1's in at least half its positions to avoid ineffective transformations that do not significantly affect the template. Number of such masks is 1.6×10^{616}



Resistance to Hacking

- Attack might consist of stealing a transformed template and trying to use it to penetrate some database by applying a transformation then attempting authentication
- Exhaustive search for the right permutation transformation of a 256 bit template requires testing on the order of $256!$ (8.6×10^{506}) candidate transformations.
- It would be easier to generate all possible templates (2^{256}) and try to match with each (actual number is less because it is not necessary for all bits to match perfectly)



Summary

- Template transformation is a powerful new tool for protecting user privacy in biometric applications
- The technique can be applied in many different scenarios, some of which have been described here
- Careful design of the template transformation implementation and infrastructure is essential

