7ZA-07

生体認証における識別器の検証機構

大崎康太†　　八槇 博史‡
東京電機大学　情報環境学部†　　東京電機大学　システムデザイン工学部‡

1. はじめに

従来の生体認証では，サーバ側に生体情報を保管しているため，情報の流出の恐れがあった．また，生体認証の一つである FIDO[1]では，サーバに生体情報を保管しないが，クライアント側の結果に依存してしまう点があった．識別器を使った検証を行うことによって，サーバへの保管無しでクライアント側の認証過程がサーバ側から確認できる検証機構を提唱する．

2. 識別器検証

システムは，登録と認証の二つのフェーズを持つ．図１は，登録フェーズの手順を示す．登録フェーズでは，利用者が利用者名を入力する．そしてクライアント側の PC の Web カメラで顔の情報を入力する．識別器を作成し，入力した顔情報を使って学習処理を行う．その後，識別器をハッシュ値に変換し，利用者名と一緒にサーバ側に送信する．

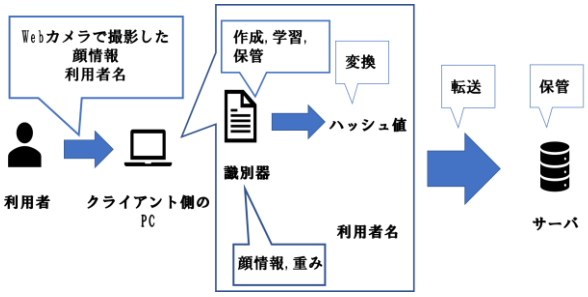

図 1　アカウントと識別器の登録

図２は，認証フェーズの手順を示す．利用者は，利用者名を入力し，PC の Web カメラを使って顔の情報を入力する．登録時に作成した識別器を使って，分類を行う．本人と分類した場合，利用者名と識別器ファイルをサーバ側に送信する．サーバ側では，送られてきた識別器ファイルをハッシュ値に変換し，利用者名とハッシュ値の組み合わせを保管情報との照合を行う．合致するアカウントがある場合，OK をそうでない場合，NG を返す．

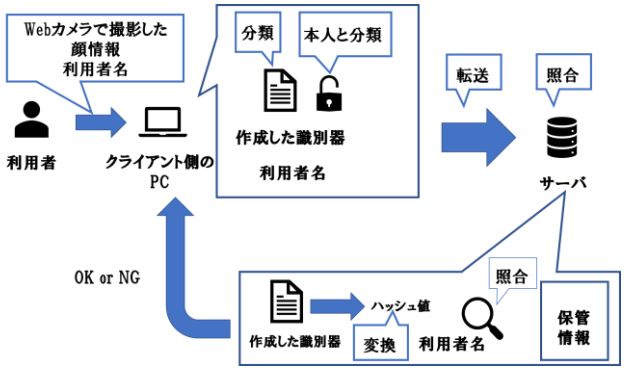

図 2　利用者と識別器の認証

登録時にデータセットを取得し，識別器を作成し署名にするため，分類の精度や，分類にかかる時間，データセットの量，質の調整が可能である．例えば，学生，一般家庭で使うような PC での認証ならば，識別器の精度が緻密でなく，学習，分類の時間が高速な識別器を作る．

3. FIDO との違い

サーバに生体情報を保管しない認証として FIDO[1]が存在する．FIDO[1]では，クライアント側で生体認証を行い，チャレンジレスポンスに

A verification mechanism with classifier in biometric authentication

† Osaki Kouta, Tokyo Denki University, School of information Environment

‡Yamaki Hirohumi, Tokyo Denki University, System Design

使う秘密鍵を取得する．その後，サーバ側が送ったチャレンジに対して秘密鍵でレスポンスを送る.しかし,FIDO[1]にも問題点が存在する．それは，チャレンジによって利用者を認証するために使う秘密鍵の取得を，クライアント側の結果だけに依存してしまう点にある．識別器検証では，利用者の認証の結果だけでなく過程を確認することが違いである.

4. 実装

4.1. データセット

OpenCV の VideoCapture[2]機能を使って,PC に搭載された Web カメラで動画を撮影し，フレームを抽出し，フレームごとの顔部分を切り出し，データセット用に成形し識別器に入力をする.

4.2. 識別器

識別器の作成，学習，保存はオープンソースソフトウェアである Keras[3]を使用した.

4.3. 実験内容と結果の定義

以下の四つの条件での正しい認証が可能か実験を行う．条件は，利用者名，ハッシュ値の組み合わせでそれぞれ正しい，誤りである場合で実験を行う．利用者名とハッシュ値が正しい組み合わせのみ認証成功であり，それ以外はエラーメッセージを出力し認証を防ぐことができれば成功である.

・利用者名　kouta
・識別器　kouta2.h5
・識別器のハッシュ値
05059ba6239ae31b23e3e7e23c31fc59d49035d3

あらかじめ上記の利用者カウントを登録しておき，利用者名，識別器ファイルの組み合わせを変えて行う.

5. 結果

表 1　実装結果

番号	利用者名	ファイル名	認証結果
1	Kouta	Kouta2.h5	成功
2	Oosaki	Kouta2.h5	失敗
3	Kouta	Osaki2.h5	失敗
4	Koute	Osaki2.h5	失敗

表 1 は,実験に使った利用者名とファイル名の組み合わせと実験結果である.利用者名と識別器のハッシュ値の二つの組み合わせを使うことでバイオメトリクス認証の欠点であるサーバ側に生体情報を保管しないという状況を成立させることができた．また，クライアント側が正しい識別器を使っているかどうか確認することができた.

6. おわりに

サーバとクライアントのハッシュ値の照合となってしまうため，ハッシュ値さえ用意できてしまうと，本人でなくともなりすましが行われてしまう可能性が存在している．今後は，識別器が固有のものであることを生かし，それが出力する結果を署名として扱いサーバ側が正しい識別器であるかどうかを判断できるようになるかを検討していきたい.

参考文献

[1] Fido Alliance. (2018 年 12 月 13 日). Fido 認証概要説明. 参照先: SlidShare: https://www.slideshare.net/FIDOAlliance/fido-83445442

[2] Opencv. (2018 年 12 月 4 日). 参照先: Opencv: https://opencv.org/

[3] CholletFrançois. (2017). Deep learning with python. マニング社.