Università degli Studi di Padova

Facoltà di Ingegneria

Tesi di laurea magistrale in Ingegneria Informatica

Un sistema per la verifica di identità basato sull'analisi del disegno di simboli grafici

"A system for user verification through drawing of symbols"

Laureando:

Fabio Masarin

Relatore:

Carlo Ferrari

Anno Accademico 2010/2011

Sommario

0. Introduzione

0.1. Obiettivo pioneristico della tesi

La sfida di questa tesi è stata quella di creare un sistema di accesso software che verifichi l'identità degli utenti basandosi su un'analisi biometrica comportamentale: in particolare l'analisi di una figura/sigla, predefinita o personalizzata, disegnata dall'utente tramite una penna digitale.

La frase precedente inizia con le parole "la sfida" perché, al momento dell'inizio di questo percorso, non si è trovata traccia di progetti analoghi già esistenti; è questo il motivo per cui il progetto ha suscitato motivazione, curiosità e fantasia nella creazione, vista la sua unicità. In alcuni casi è stato comunque tratto spunto da tecniche e metodi utilizzati in progetti di un campo simile, quello dell'analisi delle firme (handwritten signature) e della scrittura a mano libera; per questo motivo viene dedicata parecchia attenzione a questa disciplina nella prima parte di questo testo.

Gli obiettivi che si volevano ottenere dal progetto sono principalmente i seguenti:

- Universalità: Si è voluto che il metodo di verifica dell'identità sia adattabile al maggior numero di ambienti e di piattaforme; in altre parole che sia utilizzabile per vari scopi e fruibile dal maggior numero di dispositivi, PC, palmari, cellulari touch, ecc.
- Semplicità: Il sistema deve essere semplice sotto l'aspetto dell'algoritmo, in modo che richieda poche risorse di calcolo e poca memoria, favorendone la diffusione in dispositivi mobili e/o dalle scarse prestazioni; ma anche semplice per gli utenti che si troveranno ad adoperarlo, che sia quindi intuitivo e non debba richiedere troppi sforzi per l'apprendimento iniziale e per il conseguente uso quotidiano.
- Scalabilità: se possibile, il sistema non deve peggiorare in termini di prestazioni o di affidabilità all'aumentare del numero di utenti registrati.
- Sicurezza: "last but not least", il sistema deve essere sicuro e affidabile, evitando per quanto possibile l'accesso alle persone non autorizzate e garantendolo a quelle autorizzate.

Non ci dilunghiamo ora nell'approfondire questi concetti. Tutte le tematiche menzionate in questo breve paragrafo saranno riprese successivamente nel testo ed espanse a dovere, precisando anche il modo in cui gli obiettivi sono stati raggiunti.

0.2. Composizione del testo

Questo documento è composto di una prima parte contenente un'introduzione di alcuni concetti relativi la biometria e il riconoscimento della scrittura. Tale sezione non ha assolutamente lo scopo

di essere una guida esauriente e completa di tutta la materia (cosa che richiederebbe innumerevoli pagine), ma solo un'accenno ad alcuni concetti e tecniche che serviranno per una migliore comprensione delle parti successive.
Tale prima parte si concentra principalmente sul riconoscimento della scrittura (branca molto esplorata negli ultimi anni, che ha dato vita a diversi progetti e trattati a riguardo) perché è quanto ci sia di più vicino e in certi aspetti più simile al riconoscimento del disegno, che ci accingiamo a mettere in pratica tramite questo progetto, il quale può quindi essere definito "pionieristico" perché non si è trovata alcuna traccia di lavori simili. Dal riconoscimento della scrittura sarà così presa ispirazione per tecniche ed idee da applicare a questa tesi.

La seconda parte tratterà gli aspetti relativi al progetto, come le scelte intraprese o scartate e le relative motivazioni.

La terza parte spiega in dettaglio il funzionamento degli algoritmi che compongono questo sistema, i vincoli per un corretto utilizzo e le possibilità di personalizzazione.

La quarta ed ultima parte infine riporta i vantaggi ottenuti dal sistema finito, i test effettuati su di esso, la descrizione di come sono stati effettuati ed una analisi conclusiva.

1. Parte prima:
Aspetti rilevanti della biometria, definizioni e concetti

1.1. Due lati della biometria; sfide e difficoltà

La biometria, scienza che ha come oggetto di studio la misura delle variabili fisiologiche o comportamentali tipiche degli organismi, attraverso metodologie matematiche e statistiche, si compone di due diverse forme, "how you are" e "how you behave" [1].
- "How you are" è la porzione di biometria che si concentra sullo studiare caratteristiche fisiologiche/fisiche degli utenti, cioè "come essi sono", la cui presenza e univocità siano le più possibili invariate nel tempo: alcuni esempi sono dati dall'analisi dell'impronta digitale, nota per essere pressoché fissa col passare degli anni, l'analisi facciale, quella dell'iride e la più recente analisi dell'orecchio, in cui gli studi stanno attualmente dedicando parecchi sforzi perché sembra essere una delle parti del corpo umano che subisce meno mutamenti con il passare degli anni.
- "How you behave" invece studia il modo di riconoscere gli utenti sulla base di come si comportano; esempi di esperimenti in questa direzione sono l'osservazione dei movimenti degli occhi, i movimenti del mouse durante la navigazione in internet (ad esempio durata e frequenza degli spostamenti), il modo e la velocità di scrittura sulla tastiera, l'analisi della calligrafia/scrittura a mano libera e l'analisi della firma.

Senza soffermarsi su vantaggi, svantaggi e peculiarità della scienza dell'"how you are", che già da sola richiederebbe numerosi approfondimenti ed esula dallo scopo di questa tesi , ci soffermiamo nel seguito a toccare alcuni aspetti dell'"how you behave", (detto anche "behavioral biometrics") visto che il sistema progettato, che verifica l'identità dell'utente in base ad un suo disegno fatto a mano, appartiene proprio a questa seconda categoria. In particolare ci riferiremo all'analisi della scrittura manuale, che offre spunti e strumenti interessanti per lavorare sull'analisi del disegno.

Il paper [1] suggerisce un interessante schema delle ramificazioni dell'analisi della scrittura, riportato in Figura 1.

Una buona tecnica biometrica deve avere almeno queste quattro caratteristiche ([1], [2], [3]):

- Universality : ogni persona dovrebbe possedere le caratteristiche;
- Uniqueness: non si dovrebbero trovare due persone con le stesse caratteristiche;
- Permanence: le caratteristiche non dovrebbero cambiare nel tempo;
- Collectability: le caratteristiche dovrebbero essere facilmente presentabili a un sensore, quantificabili e comparabili.

Per quanto riguarda l'analisi della scrittura, val la pena di spendere qualche parola riguardo queste quattro caratteristiche, perché sono meno scontate rispetto a quanto potrebbero esserlo nelle tecniche di analisi "how you are". Infatti [1], per quanto riguarda l'Universality non è detto che

tutte le persone sappiano scrivere, una parte della popolazione mondiale è ancora analfabeta. Sull'Uniqueness, è teoricamente possibile trovare due persone con una calligrafia pressoché uguale. Rispetto alla Permanence, sappiamo bene che la calligrafia della persona varia con l'età; come soluzione a questo problema è sufficiente eseguire periodicamente negli anni delle nuove procedure di Enrollment; ma essa varia anche in base allo stato emotivo e psicologico dell'individuo nel momento in cui scrive, e ciò rappresenta la difficoltà principale, come vedremo nel seguito. Infine sulla Collectability non ci sono particolari difficoltà, in quanto la scrittura può essere acquisita tramite tavolette grafiche, tablet PC o, in caso di campioni di testo scritti su carta, possono essere digitalizzati tramite scanner.

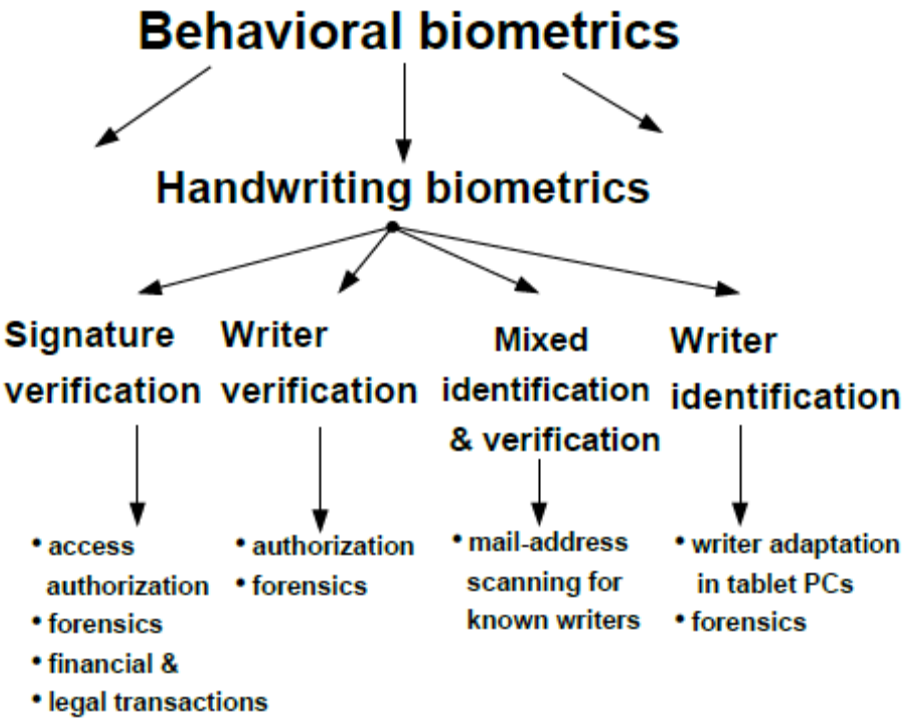

Figura 1 - Alcune delle attuali branche dell'analisi della scrittura

Lo studio dell'"how you behave" è utile perché ha un interessante vantaggio: il fatto che solitamente dà luogo a tecniche non invasive. Si pensi infatti alla differenza tra una tavoletta grafica su cui apporre la propria firma e uno scanner dell'iride, e si provi ad immaginare la maggior diffidenza con cui sarebbe accettato il secondo dei due da un gruppo di utenti, magari abituati ad usare una semplice password fino al giorno prima per accedere ai propri sistemi.

Dall'altro lato la difficoltà principale nell'osservazione del comportamento, come già accennato, è che la calligrafia non è sempre identica come lo è un l'impronta digitale o l'iride, ma anzi, varia anche considerevolmente a seconda dello stato psicologico in cui si trova l'utente, l'eventualità di situazioni di stress, la stanchezza, la posizione della mano o del braccio rispetto alla superficie di appoggio, ecc. Secondo [1] inoltre, durante la scrittura la persona manda dei "motor-control patterns" che sono compensati dai muscoli che agiscono da filtri passa-basso e generano delle

variabilità, come ad esempio dei tremori, che sono molto più dipendenti dallo stato della persona anziché dalla sua identità.

Quanto detto finora però non deve spaventare, perché sul riconoscimento della scrittura ci sono ancora dei punti interessanti da menzionare che giocano a favore. Il primo è la cosiddetta "sequencing variability" [1]: ossia la consapevolezza che per scrivere uno stesso simbolo, come ad esempio una lettera dell'alfabeto, ci sono vari modi determinati dall'ordine con cui una persona è abituata a disegnare i tratti (per un esempio di questo si veda il paragrafo 2.1.) . Il secondo è la conoscenza del fatto che la calligrafia e la dimensione della scrittura è spesso dipendente da alcuni aspetti strutturali della mano e dell'individuo, come ad esempio la dimensione relativa delle ossa carpali, la forza esercitata nella presa della penna e nella pressione della stessa sulla superficie di scrittura, l'essere destri o mancini, la muscolatura della mano, eventuali problemi motori o periferici della persona, proprietà del sistema nervoso, ecc. ([4]). Infine anche l'educazione dell'individuo, intesa come influenze culturalmente trasmesse nell'insegnamento della scrittura (la presa della penna, la forma da dare alle lettere) influisce sulla calligrafia personale.

1.2. Cenni su "User Identification" e "User Verification"

Precisiamo ora la differenza tra i concetti di User Identification e User Verification:

- Identification : Dato un campione biometrico, indipendentemente dalla caratteristica osservata (analisi facciale, dell'impronta digitale, dell'iride, analisi comportamentale, analisi della scrittura), si parla di User Identification quando il sistema a partire da tale campione, senza altre informazioni dichiarate, riconosce e quindi fornisce in output l'identità (presunta) dell'utente corrispondente tra le N identità contenute nel bagaglio informativo di tale sistema.
- Verification: Si parla di User Verification quando dato ancora un campione biometrico ed una dichiarazione di identità (del tipo "Io sono l'utente X e sto fornendo questo campione"), il sistema si occupa di stabilire se la dichiarazione è vera o falsa, ossia se il campione appartiene davvero all'utente X o se si tratta di un impostore che si sta fingendo qualcun altro per accedere a sistemi/informazioni non autorizzate.

In altre parole, stando alla definizione in [1], Identification significa trovare una sequenza di azioni in una data lista (1:N), mentre Verification significa verificare dell'identità di un individuo tramite un confronto 1:1.

Il progetto sviluppato nel corso di questa tesi, come vedremo nel seguito, è un progetto di User Verification perché si dichiara dapprima il proprio nome, dopodiché si fornisce il campione biometrico necessario alla verifica dell'identità dichiarata, dove il campione biometrico fornito è, come ormai sarà chiaro, una figura disegnata a mano libera.

1.3. Caratteristiche di studio per il riconoscimento basato sulla scrittura

Dopo le considerazioni di carattere generale, è giunto il momento di abbassarsi un po' di livello e di cercare quali potrebbero essere le caratteristiche (nel seguito dette alternativamente anche "feature") che devono essere ricavate per poter costruire il sistema.
Stando alle trattazioni e agli esperimenti in [1] [5] e [6] (il quale invece tratta un sistema di User Identification basandosi sulle realizzazioni di un cerchio usando il mouse) le feature interessanti nello studio della scrittura sono le seguenti:

- Traiettorie X e Y del disegno nel tempo, raccolti come sequenze temporali x_n e y_n;
- Segnale di pressione della penna sulla superficie, anch'esso sottoforma di sequenza temporale p_n;
- Segnale di altezza della penna rispetto alla superficie ϕ_n (parametro che si rileva quando la penna non è a contatto con il piano ma si trova entro la distanza di rilevamento, tipicamente 5 millimetri dalla superficie);
- Segnale dell'angolo di inclinazione della penna γ_n (azimuth) rispetto alla superficie;
- La misura T del tempo impiegato per completare la scritta/figura.

Alcune di queste ed altre feature verranno utilizzate nel sistema proposto in questa tesi; la loro trattazione sarà integrata ed espansa nel paragrafo 2.4.

Altri metodi interessanti per lo studio della calligrafia, (mostrati in [7]), sono la ricerca di primitive frequenti nella scrittura manuale comparando la variabilità in tali primitive per identificare l'utente ed altresì lo studio dei punti di massima e minima velocità durante il disegno di tali primitive.
Per maggiori approfondimenti sull'uso di queste tecniche e sulle possibili funzioni di calcolo della somiglianza si rimanda a [7]; in questo testo non saranno trattate perché non utilizzate nel sistema progettato in quanto ritenute più adatte alla scrittura testuale piuttosto che al disegno di figure semplici.

Altri metodi ancora si basano sull'uso di HMM (Hidden Markov Models) [5] o sull'estrazione di feature estese calcolate come funzione delle feature di base elencate nel precedente elenco [5]. Anche in questo caso non tratteremo a fondo tale argomento, in quanto la trattazione di questo paragrafo ha il solo scopo di dare un'infarinatura di concetti utili alla comprensione del seguito del testo.

In sostanza, alla luce di quanto detto, va tenuto presente che proprio per via della variabilità del comportamento umano e dello stato psicologico dell'individuo nei diversi momenti in cui lo si analizza, è impossibile raggiungere la perfezione, tanto più nella "how you behave" di quanto accada nella "how you are".

1.4. Cenni sul confronto di prestazioni e indici di valutazione

Il confronto tra metodi di riconoscimento biometrico è talvolta difficile per la loro diversità intrinseca. Spesso, anche in metodi che analizzano lo stesso parametro biometrico (come ad esempio due metodi di riconoscimento facciale, due metodi di riconoscimento della scrittura) ci sono grandi differenze sul numero di variabili, sulle feature in gioco, sulle risorse di calcolo richieste,ecc.
Una delle più grosse differenze si trova nello sforzo necessario a costruire un buon sistema di training. Ad esempio un metodo può essere teoricamente molto migliore di un altro in termini di precisione nel riconoscimento, ma per funzionare deve aver un grande database di campioni alle spalle oppure richiedere procedure lunghe o complicate per l'apprendimento delle caratteristiche biometriche degli individui e quindi, paradossalmente, riscuotere meno successo di un altro metodo meno preciso, ma dal training più immediato o meno invasivo.

Nonostante queste difficoltà comunque, esistono degli indici per valutare e paragonare, le tecniche di riconoscimento biometrico. Spesso vengono usati FAR, FRR e EER [3,8].

- FAR (False Acceptance Rate): Questo indice misura la probabilità che il sistema associ erroneamente un input biometrico fornito da un utente X non autorizzato ad un utente Y autorizzato. Più brevemente la probabilità che il sistema accetti un utente non autorizzato.
- FRR (False Rejection Rate): Questo indice misura la probabilità che il sistema consideri un input biometrico come non autorizzato quando invece lo è. In altre parole è il contrario del FAR, ossia la probabilità che un utente autorizzato sia rifiutato.
- EER (Equal Error Rate): rappresenta il valore per cui FAR e FRR coincidono in un determinato sistema.

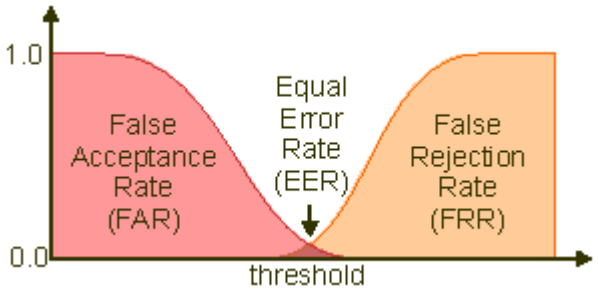

Figura 2 - Scelta dei parametri nella regolazione di FAR e FRR; la soglia EER rappresenta un buon compromesso (figura tratta da [8])

Per calcolare gli indici FAR e FRR di un sistema sotto test, è necessario fare numerose prove e calcolare la percentuale di insuccessi che ricadono nelle due categorie di cui gli indici fanno parte. Per abbassare uno dei due indici, bisognerà agire sui parametri propri del sistema, i quali ovviamente variano da caso a caso e quindi non sono univoci. Quello che invece accomuna la fase di taratura dei parametri è che al calare di uno dei due indici, l'altro aumenta.
La decisione su quanto sacrificare uno a favore dell'altro va presa a seconda del campo di applicazione. Ad esempio in un ambiente dove è richiesta la massima sicurezza si può far in modo che il FAR sia ridotto al minimo tollerando un aumento del FRR; in tal modo le persone non autorizzate non riusciranno ad entrare, ma bisogna accettare il fatto che con un FRR più alto si

dovrà talvolta ripetere la fase di accreditamento anche se si è autorizzati. Dall'altro lato, in un ambiente dove non si vuole perdere tempo a patto di tollerare che qualche utente non autorizzato possa essere ammesso (magari bloccandolo con qualche successiva procedura di accertamento) si può agire sui parametri in modo da abbassare il FRR ed alzare il FAR.

Nel riconoscimento delle impronte digitali, il riconoscimento biometrico più approfondito fino ad oggi e che offre probabilmente la maggiore precisione, si richiedono tipicamente dei valori di FAR inferiori a 0,01% e di FRR inferiori a 0,1% [9]. Questi valori sono oggettivamente molto bassi, proprio perché il riconoscimento delle impronte digitali è il settore della biometria che ad oggi offre le prestazioni migliori; tipici valori di FAR e FRR in altri metodi di riconoscimento sperimentali si attestano attorno al 5-10%. Si vedano ad esempio [1], [5], [6], [7], [10].

Un tradeoff spesso accettabile e usato anche per il confronto tra metodi diversi, è quello di fare in modo che il FAR eguagli il FRR ; in tal caso la probabilità assume come già detto il nome EER e riducendosi ad un solo valore è più idonea ai confronti.

Ancora una volta non va dimenticato che l'indice EER da solo non è sufficiente per una comparazione esaustiva di metodi di riconoscimento biometrico. Va anche considerata l'invasività dei metodi, il tempo/difficoltà necessari ad acquisire i campioni necessari all'Enrollment e alla base di dati, le risorse di tempo e calcolo richieste dall'algoritmo, il modo in cui il metodo viene accettato dagli utenti (cioè quanto è user-friendly), e così via. Caratteristiche che talvolta non possono essere quantificate ma valutate solo qualitativamente, se necessario caso per caso, in relazione all'ambiente di utilizzo.

Inoltre il calcolo di EER è molto più difficile del calcolo di FAR e FRR; sia perché per ottenere risultati precisi bisognerebbe calcolarlo con un test set rappresentativo ed infinito, ma soprattutto perché per confrontare metodi diversi tramite EER bisognerebbe usare lo stesso test set su tutti i metodi sotto analisi [8].

Relativamente al caso specifico di questo progetto quindi, come vedremo nella quarta parte del documento, non possiamo calcolare l'indice EER sia per la quantità limitata di persone che hanno effettuato il test sia perché, come già discusso in precedenza, essendo un progetto unico nella sua unica natura non esistono altri metodi simili a cui dare gli stessi test set in input. Saranno calcolati quindi solo i parametri FAR e FRR ottenuti.

2. Parte seconda: Introduzione al progetto

2.1. Come nasce l'idea

L'idea alla base di questo progetto è nata durante la lettura di alcuni paper riguardanti lo studio delle firme a mano libera per verificarne l'autenticità.
Parecchi sforzi sono stati dedicati nel tempo in questo campo , anche per motivi legali, e si distinguono principalmente nello studio di documenti "offline" e di documenti "online" [7]. Un documento "offline" è una scansione di un documento scritto a mano, nel caso specifico una firma già fatta; un documento "online" invece dispone anche dell'informazione temporale del processo di scrittura e degli eventi di pen-up e pen-down. Questo è possibile con strumenti come tavolette grafiche o tablet PC, la cui recente diffusione sta rendendo più popolare l'analisi di documenti online.

Ebbene, durante la lettura di tali paper, è nata l'idea di utilizzare una analisi del genere per applicarla ad un sistema di accesso a sistemi generici, che potrebbe venir applicata in svariati ambiti, ovunque siano disponibili dispositivi touch (tavolette grafiche, tablet PC, smartphone touch, e quant'altro). Vista quindi l'ambizione di renderlo un sistema generico e il più dinamico possibile, si è voluto che tale sistema fosse adoperabile anche via internet; il più semplice esempio che ci può venire in mente è, dato un qualunque sito internet a cui oggi accediamo tramite nome utente e password, accedere un giorno allo stesso sito facendo invece un disegno a mano; il che, visto nell'ottica dei dispositivi touch mobili sempre più diffusi e potenti, sarebbe senz'altro più comodo.
Subito dopo aver avuto questo pensiero, si è accesa anche un'altra lampadina: quella che dalla parola internet porta alla parola privacy. Più precisamente, se sulla base di questo principio ogni utente utilizzasse la propria firma come chiave di accesso, la cosa potrebbe probabilmente funzionare, ma sarebbe molto rischiosa in termini di privacy e di sicurezza; fare la propria firma spesso, in luoghi potenzialmente visibili al pubblico, e trasmetterla via internet, è qualcosa che a una prima analisi sarebbe meglio evitare. Ecco così che l'idea di utilizzare un qualche disegnino anche piuttosto semplice o (se proprio manca "la vena artistica") una qualche sigla che si è usi o pratici fare, è sembrata ancora più appropriata allo scopo.

Spinto da questo pensiero, la lettura dei paper a disposizione è continuata nell'obiettivo di approfondire come fino ad ora sono state impostate le analisi delle firme, per vedere cosa se ne poteva trarre per dar luce a questa nuova idea. La cosa è sembrata ancora più fattibile leggendo il paper [1], il quale mostra che per disegnare una semplice lettera E maiuscola, ci sarebbero ben $4!*2^4 = 384$ modi diversi (dovuti al fatto che i quattro tratti che compongono la lettera possono essere disegnati da sinistra(o alto) a destra(o basso) e viceversa dando luogo a 2^4 possibilità, e l'ordine di disegno dei 4 tratti può essere scelto in 4! modi). Nella pratica le persone usano molte

meno di queste 384 possibilità (perché molte sono poco pratiche e poco sbrigative), forse le più diffuse sono solo 4 o 5, però questo ha suggerito come sia possibile, se già per disegnare una semplice "E" c'è tutta questa variabilità, ottenere grandi possibilità di differenziazione dal disegno di simboletti poco più complessi di questo; senza contare il fatto che applicando questo concetto ad un sistema di sicurezza, si può sempre chiedere ad un utente di disegnare la lettera "E" nel modo più strano che gli viene in mente, ed ecco che possono così tornare in gioco tutte le 384 possibilità.

In altri paragrafi verrà poi discusso con maggior dettaglio, quali aspetti psico-motori e del disegno stesso vengono osservati per verificare l'identità dell'utente, ed ancora quali requisiti sono indispensabili e quali piattaforme possono usufruire di una tecnica simile.

2.2. Architettura del sistema

Questo capitolo delinea in modo sintetico l'architettura del sistema che verrà creato. E' interessante far notare che, senza entrare in dettagli che si troveranno nel seguito, l'intero sistema si baserà su due sole "procedure", la procedura di Enrollment, cioè quella che registra i dati di un nuovo utente per imparare a distinguerne i comportamenti, e la procedura di Login, cioè quella utilizzata ad ogni tentativo di accesso al sistema. A tali due procedure sono associati i due algoritmi principali del sistema, che saranno descritti in modo dettagliato nei paragrafi 3.2. e 3.3. Lo schema a blocchi di Figura 3 mostra in modo puramente qualitativo ciò che viene fatto da tali due algoritmi, e quali dati sono sfruttati al momento della verifica di identità.

La procedura di Enrollment salva nel database del sistema i "Campioni di Enrollment", cioè 10 registrazioni della figura chiave scelta e disegnata da un generico utente X che chiede di essere ammesso al sistema; allo stesso momento, da questi 10 campioni sono estratte anche delle tolleranze appropriate per tale utente.
L'algoritmo di Login, ad ogni esecuzione, riceve un ulteriore "Campione di Login" il quale viene dato come input ad un algoritmo che calcola un punteggio di somiglianza tra esso e i 10 campioni di enrollment, aiutandosi con le tolleranze disponibili e con la scelta delle feature da analizzare nei confronti. Vi è infine una soglia di tolleranza fissa (e quindi regolabile solo dal progettista) che decide, una volta ottenuto il punteggio di somiglianza, quale dev'essere il livello sopra il quale accettare l'utente nel sistema. La soglia è attualmente pari al 70%, come vedremo nel seguito del testo.

La presenza del simbolo della chiave inglese sulle componenti "tolleranza" ed "estrazione feature" sta ad indicare il fatto che quelli sono i parametri regolabili/personalizzabili da un eventuale amministratore di sistema. Le modalità per farlo saranno espanse nel paragrafo 3.4.

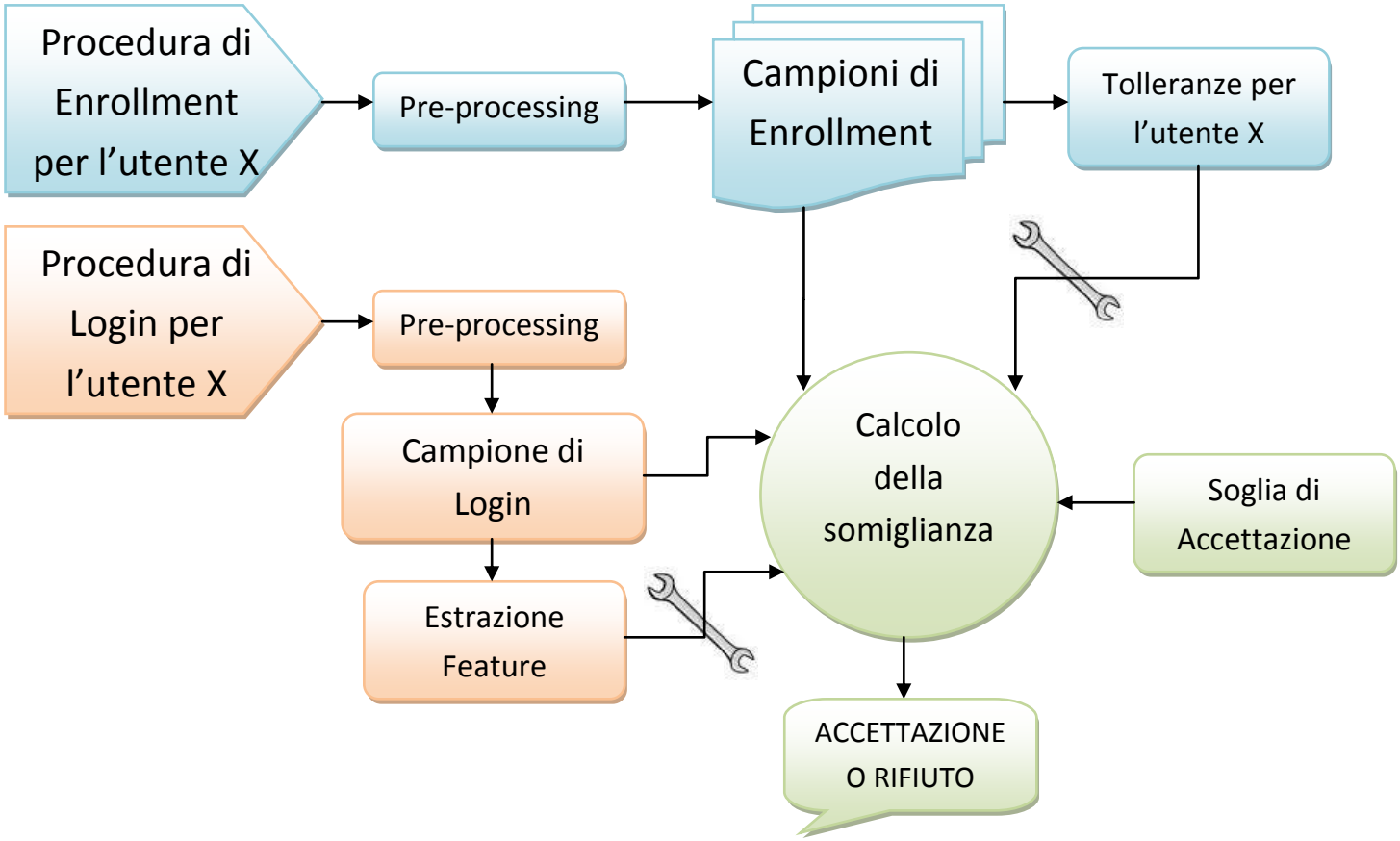

Figura 3 - Diagramma dell'architettura del sistema

2.3. Strumenti impiegati per lo sviluppo

2.3.1. Hardware: Il Tablet PC impiegato

L'attrezzatura hardware necessaria per lo svolgimento di questa tesi è limitata a un Tablet PC convertibile, modello Fujitsu-Siemens Lifebook T4010D , equipaggiato con Windows XP Tablet PC edition SP3. L'aggettivo convertibile significa che può essere aperto ed utilizzato come un normale notebook, oppure può essere richiuso su sé stesso dopo aver ruotato lo schermo in modo da essere comandato solo tramite penna su schermo. Senza soffermarsi sulle caratteristiche di questo PC portatile (dell'anno 2005 e con una comune CPU Intel Centrino 1.8 Ghz) , possono risultare

Figura 4 - Tablet PC convertibile Fujitsu Siemens T4010D

utili alcuni cenni sul suo digitalizzatore, cioè l'hardware che fornisce al proprio software la posizione della penna rendendo un qualunque portatile un Tablet PC.

Esistono attualmente 4 tecnologie di digitalizzatori per creare schermi touch screen (tratto da [11]):

- Digitalizzatore passivo resistivo: Posto sopra il pannello LCD, questo digitalizzatore è sensibile a qualsiasi cosa si prema sullo schermo. A seconda della qualità può essere capace o meno di riconoscere la pressione esercitata, mentre in nessun caso è capace di riconoscere oggetti che si trovano in prossimità dello schermo o che lo sfiorano semplicemente. Da un punto di vista fisico è composto da due strati di materiale conduttivo che, nel momento in cui un oggetto viene premuto sullo schermo, entrano in contatto riuscendo a determinare la posizione dell'oggetto. Scrivere su uno schermo dotato di digitalizzatore passivo resistivo è possibile ma nella maggior parte dei casi innaturale, perché bisogna premere in modo considerevole stando allo stesso tempo attenti a non appoggiare il polso sullo schermo; questa tecnologia non è inoltre in grado di riconoscere la posizione in modo preciso (inferiore al millimetro). Non è quindi adatto per l'uso con la penna, è però diffuso negli UMPC, nei Touch Tablet PC economici e in piccoli dispositivi come smartphone o i navigatori satellitari in quanto si può usare con le dita ed è più economico di altri digitalizzatori.
- Digitalizzatore passivo capacitivo: è la tecnologia che viene utilizzata nei touchpad ma che è diventata famosissima con l'avvento dell'iPhone. Posto sopra il pannello LCD, questo digitalizzatore riconosce la presenza di oggetti anche multipli che sfiorano o toccano lo schermo, utilizzando la misura della loro capacità elettrica. Materiali isolanti o anche le dita coperte da guanti non sono quindi percepiti. E' capace di riconoscere la posizione di una penna ma non la pressione esercitata se non in modo poco preciso. Non è particolarmente indicato per l'uso a penna in quanto non permette di appoggiare il polso sullo schermo, prima dell'avvento dell'iPad era poco sfruttato in quanto più complesso e costoso del resistivo.
- Digitalizzatore attivo elettromagnetico Wacom: è l'ideale per chi vuole scrivere. Composto da una griglia che crea un piccolo campo elettromagnetico verso l'esterno e da una particolare penna che si attiva con questo campo, questo digitalizzatore viene montato dietro il pannello LCD ed offre una elevata precisione di scrittura unita ad una sensibilità a non meno di 256 livelli di pressione. Scrivere con un digitalizzatore Wacom è facile e la penna può supportare pulsanti per funzioni aggiuntive come ad esempio la funzione gomma o "tasto destro". Il digitalizzatore è sensibile alla sola penna Wacom e non alle dita permettendo così di appoggiare il polso sullo schermo e scrivere in modo completamente naturale.
- Digitalizzatore DuoSense N-Trig: questo particolare digitalizzatore unisce le funzionalità di un digitalizzatore attivo elettromagnetico a quelle di un digitalizzatore passivo capacitivo,

integrandole in un unico pannello. Perfetto per l'uso a penna, integra la possibilità di essere utilizzato anche con un massimo di quattro dita contemporaneamente.

Il Tablet PC utilizzato in questa tesi monta un digitalizzatore attivo Wacom; esso fornisce tramite il suo driver, ogni secondo, 133 letture della posizione della penna (133 Hz).

Da ogni lettura si ricava, attraverso l'analisi dei campioni restituiti:

- Le coordinate X e Y della posizione della penna sullo schermo, espresse in decimi di millimetro; per passare da tale unità di misura ai quella dei pixel bisogna moltiplicare per 25,88 che è il relativo fattore di conversione. Questi dati sono rilevati anche quando la punta della penna non è in contatto con lo schermo,ma si trova ad una distanza non superiore ai 5 millimetri.
- La pressione applicata dalla penna sullo schermo. Il valore ottenuto in questo caso non ha unità di misura, ma è un numero puro che varia nell'intervallo da 0 (nessuna pressione) a 255 (pressione massima) nel caso sia installato il driver Wacom v. 1.04 (quello originariamente fornito con il Tablet PC); l'intervallo va invece da 0 a 1023 nel caso si sia installato il driver più aggiornato (la versione 5.05 al momento in cui scrivo, scaricabile dal sito ufficiale Wacom [12]).

2.3.2. Software: IDE e SDK ausiliario

Dal lato software invece , si è scelto di sviluppare il programma nella piattaforma Visual Basic .NET; questo sia per la familiarità già acquisita con questo linguaggio di programmazione, sia per la migliore "vicinanza" al pacchetto di comunicazione con il driver del digitalizzatore; per quanto riguarda i digitalizzatori dei Tablet PC non esistono ancora driver sufficientemente stabili per linux.

Si è così scaricato il pacchetto Visual Studio .NET 2010 dalla piattaforma MSDNAA, tramite licenza gratuita per studenti dell'Università di Padova. Fatto questo, per poter usufruire dei metodi per il collegamento al driver della penna, è necessario installare anche il Microsoft Tablet PC Software Development Kit (per il tablet PC di riferimento si trova la versione 1.7 per Windows XP all'indirizzo [13]). Questo SDK permette di includere le rispettive librerie nel progetto di Visual Studio e di poter usufruire delle classi e dei metodi forniti. Per stabilire il collegamento con le SDK, dalla finestra principale di Visual Studio si deve entrare nel menu Progetto->Aggiungi Riferimento (Add reference), cliccare su sfoglia e importare: "c:\programmi\Microsoft Tablet PC Platform SDK\Include\Microsoft.Ink.dll".

Installato tutto il software necessario, si è passati a ricercare come ottenere i dati della penna al livello più basso possibile. Il SDK sopra citato infatti, fornisce delle interessanti classi di alto livello che permettono il preprocessing dell'input oppure di implementare nei propri programmi funzioni avanzate con il minimo sforzo, come ad esempio il riconoscimento del testo scritto con la penna, il

riconoscimento di “hand gestures”, ecc. Per dettagli su queste funzionalità si veda la libreria MSDN agli indirizzi [14]e [15].
Ciò però non è quanto occorreva per questo progetto: si richiedeva infatti di ottenere i dati “grezzi” dalla penna, per stabilire semplicemente le coordinate in cui si trova e la pressione impiegata. La classe per far questo si chiama RealTimeStylus [16], e la sua diffusione e documentazione è paradossalmente meno folta di quelle delle classi a più alto livello. Interessanti guide per praticare i primi passi con tale classe si trovano in [17] o [18]; la lettura di queste guide ha permesso così di scrivere il primo programma che ricevesse input dalla penna, dal quale poi è continuato il resto del lavoro.

2.4. Feature scelte: forma, vantaggi e motivazioni

In questo paragrafo tratteremo in maggior dettaglio le feature scelte per l’analisi del disegno in questo progetto, già parzialmente citate nella prima parte del testo, precisando per ognuna di esse le motivazioni per le quali è stata scelta ed impiegata, e il modo in cui i rispettivi dati sono forniti al sistema (variabile singola, array, matrice,ecc).

- ✓ Tempo impiegato. La prima feature inclusa è l’indicazione del tempo impiegato a completare la figura. Questa quantità include il solo tempo in cui la penna è a contatto con la superficie di scrittura (in questo caso, visto che si è utilizzato un tablet PC, tale superficie è lo schermo, indicato così nel seguito), in altre parole la somma dei tempi impiegati per il disegno dei singoli tratti. Esemplificando, nel disegno di una figura composta da due tratti, se si disegna il primo tratto in un tempo T_A=0,7 secondi , poi si stacca la penna per un tempo T_P = 2 secondi ed infine si disegna il secondo tratto in un tempo T_B = 1 secondo, il tempo impiegato $T = T_A + T_B$ = 1,7 secondi; quei due secondi di “pausa” non sono considerati.
  Questa feature è fornita sottoforma di numero intero, la cui unità di misura non è espressa in secondi, ma in numero di campioni ricevuti dall’hardware della penna (per maggiori dettagli sulla composizione dei campioni rilevati e sulla loro frequenza, si veda il precedente paragrafo 2.3.1.).
- ✓ Numero dei punti di attacco impiegati per il disegno. Questa feature conta quante volte viene appoggiata sullo schermo la penna durante il disegno della figura; ciò equivale al conteggio di quanti tratti l’utente impiega per disegnare una certa figura. Data la molteplicità di modi con cui le figure possono essere disegnate, è necessario che l’utente scelga un’unica sequenza di tratti e la mantenga nel tempo (si vedano i paragrafi 3.1 e 4.1 per comprendere i vantaggi in termini di sicurezza che questa condizione offre). Questa feature è fornita sottoforma di numero intero, attualmente variabile tra 1 e 10, visto che il sistema memorizza al massimo 10 punti di attacco (considerati più che sufficienti per il disegno di una semplice figura o sigla).

- ✓ Posizione dei punti di attacco. Per ciascuno degli al più 10 punti di attacco P_i, come definiti poc'anzi, il sistema memorizza alcune informazioni, che sono:
  - Le coordinate X_i e Y_i in cui si è verificato, cioè le coordinate in cui l'utente ha appoggiato la penna all'i-esimo punto di attacco;
  - L'istante temporale T_i in cui si è verificato, espresso con la stessa unità di misura (numero di campioni ricevuti) con cui è espressa la feature di tempo impiegato.

  Questi dati sono quindi forniti in una matrice di dimensioni 10x3, dove 10 sono i punti di attacco massimi, e 3 sono le informazioni intere ad esso associate. In caso un punto di attacco non si sia mai verificato, la corrispondente riga della matrice conterrà tre valori '-1'.
- ✓ Segnale temporale delle coordinate X(n) : Questa feature memorizza la traiettoria effettuata dalla penna rispetto all'asse X durante l'effettivo disegno;anche in questo caso si considera la posizione della penna unicamente quando essa è appoggiata allo schermo. I dati di questa feature sono forniti in un vettore X(n), composto da un numero di elementi pari al valore della feature "Tempo impiegato" (1<=n<=T), che contiene in ogni cella la posizione sull'asse X della penna all'n-esimo istante, posizione espressa da numeri in virgola mobile variabili tra 0 e 1, dove 0 rappresenta l'estremità sinistra della zona adibita al disegno della figura chiave ed 1 la rispettiva estremità destra. Questo vettore subisce poi una serie di normalizzazioni ed elaborazioni, che saranno dettagliate nel paragrafo 3.3. riguardante l'algoritmo di Login.
- ✓ Segnale temporale delle coordinate Y(n). Quanto appena detto per la feature precedente vale anche in questo caso, stavolta rispetto alle coordinate dell'asse Y.
- ✓ Segnale temporale della pressione impiegata dalla penna Z(n). Anche in quest'ultimo caso vale quanto detto per i segnali delle coordinate X e Y, nelle stesse modalità, dove i valori nel vettore sono ancora compresi tra 0 (pressione nulla) e 1 (pressione massima). Si noti che siccome la registrazione della posizione avviene solo quando la penna è effettivamente sullo schermo, il valore 0 non sarà mai presente nel vettore.

A scopo chiarificativo, la Figura 5 mostra la rappresentazione grafica di vettori di Coordinata X e di Pressione per due diversi disegni fatti a mano; si notino la differenza delle curve e la presenza dei punti di attacco della penna, indicati con dei piccoli cerchi di colore verde nel pannello di sinistra, mentre nei grafici si trovano in corrispondenza dei punti di discontinuità.

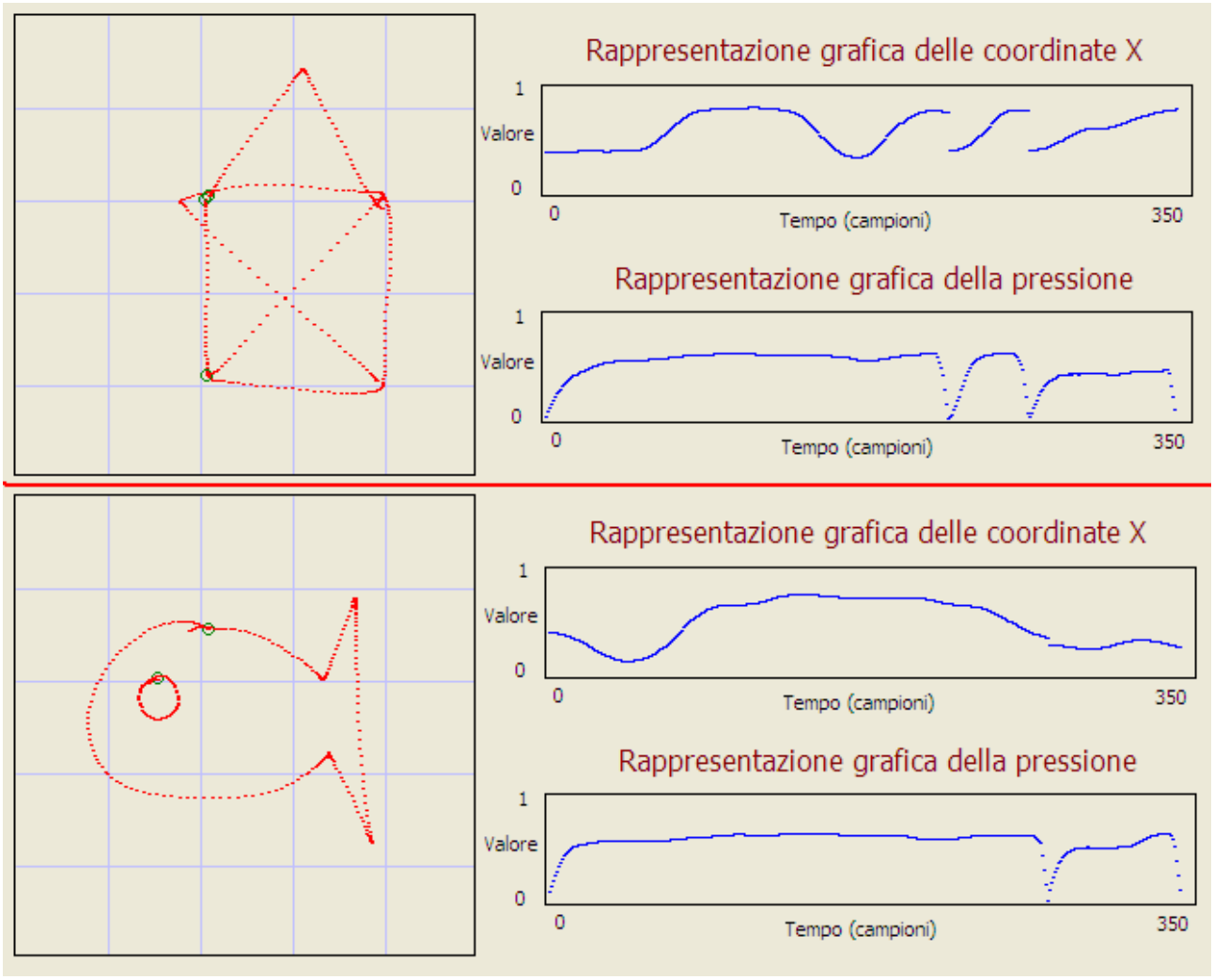

Figura 5 - Rappresentazione grafica del contenuto dei vettori di coordinate e di pressione per una coppia di disegni; i valori sull'asse verticale sono normalizzati da 0 ad 1, che rappresentano rispettivamente per le coordinate l'estremo sinistro l'estermo destro oppure la pressione minima e la pressione massima.

I vantaggi forniti dalle feature scelte sono di non poco conto:

- Registrare e confrontare il tempo impiegato nella produzione di uno specifico disegno permette di distinguere utenti che per abitudine, modo di disegnare, e le altre peculiarità già descritte nel paragrafo 1.1., disegnano una stessa figura o sigla in tempi diversi.
- Registrare e confrontare il numero di punti di attacco impiegati fornisce l'importantissima possibilità di distinguere se un utente ha disegnato la stessa figura con stessa posizione, geometria, proporzione e ordine di tratti di come l'ha fatta in fase di Enrollment. Questo aspetto a dir poco cruciale, che inizialmente può sembrare un limite perché costringe un utente a ricordarsi come aveva fatto il disegno (ma del resto anche in un sistema tradizionale ad accesso con password ogni utente deve ricordarsi la propria), dà invece una grande spinta alla sicurezza, perché permette di bloccare eventuali aggressori che tentano di entrare in un account altrui conoscendo qual è la figura chiave, ma non sapendo esattamente dove deve essere posizionata e in che ordine vanno fatti i tratti. (si veda più avanti nel testo la Figura 9).

- Registrare e confrontare le traiettorie sugli assi X e Y, banalmente, serve a distinguere la geometria delle figure, distinguendole una dall'altra in caso di figure che potenzialmente potrebbero avere gli stessi punti di attacco ma traiettorie diverse.
- Registrare e confrontare la pressione della penna è di nuovo una feature che permette di discernere gli utenti a livello comportamentale, come nel caso del tempo impiegato.

2.5. Feature scartate: motivazioni

In questo paragrafo elencheremo invece le feature che sono state scartate e non incluse in questo progetto, alcune delle quali già parzialmente citate nella prima parte del testo , precisando per ognuna di esse le motivazioni che hanno portato al mancato impiego.

- X Segnale temporale dell'inclinazione della penna. questa feature non è stata inclusa nel progetto di questa tesi per tre motivi:
  - in primo luogo il digitalizzatore del tablet PC utilizzato per questa tesi non è in grado di rilevare l'inclinazione dell'asse della penna rispetto allo schermo (cosa che invece sono in grado di fare alcune tavolette grafiche), rendendo di fatto impossibile la sperimentazione con l'hardware a disposizione;
  - in secondo luogo anche se fosse stato possibile leggere tale feature, sarebbe stata poco aderente all'obiettivo di universalità del progetto (si vedano par. 0.1 e 4.1.1), in quanto nei dispositivi mobili sensibili alle dita anziché alla penna non ha senso parlare di "inclinazione del dito rispetto allo schermo";
  - in terzo luogo (last but not least) stando alle sperimentazioni in [5], relative allo studio dell'autenticità della firma, le prestazioni peggiorano passando da un EER pari al 4,54% quando la feature non è considerata, ad un EER pari al 4,84% considerandola.
- X Segnale temporale dell'altitudine della penna dallo schermo. Questa feature, che considera la distanza della penna dallo schermo unicamente nei momenti in cui la penna non è a contatto ma si trova entro il raggio di rilevamento dello schermo (tipicamente 5 mm), è stata scartata perché:
  - Anche questa feature non è rilevabile dal tablet PC: il tablet è capace di avvertire la vicinanza della penna entro 5 mm di distanza dalla superficie dello schermo ma non è in grado di stabilire a che distanza si trova;
  - Come già premesso è una feature che esiste solo nei momenti in cui la penna stacca dallo schermo. Siccome si è deciso di considerare significativo solo ciò che succede quando la penna è in contatto, il considerare questa feature non aggiungerebbe informazione utile e per di più, penalizzerebbe delle realizzazioni della figura chiave in cui, per qualche motivo come una momentanea distrazione, l'utente ha sospeso il disegno della figura chiave tra un tratto e il successivo;

- Come nel caso precedente, portando la feature su altre piattaforme non esiste la "distanza del dito dallo schermo di un dispositivo touch";
- Infine, sempre secondo gli studi di [5], l'EER peggiora passando da un 4,54% a un 6,28% quando si aggiunge l'altitudine della penna al set di feature considerate.

X Segnali avanzati derivati dalla composizione dei segnali base. Di questi segnali ne esistono parecchi, ed essendo composizione di funzioni con i segnali base ne possono nascere a volontà. Alcuni esempi, tratti da [5] sono le derivate dei segnali, l'accelerazione della penna,la velocità angolare, ecc.
Un esperimento è stato condotto durante lo svolgimento della tesi calcolando il segnale della derivata delle coordinate X e Y ma a seguito di alcuni test non ha prodotto vantaggi rispetto alla considerazione delle semplici coordinate X e Y, al che è stato scartato. Inoltre si è preferito non entrare nel calcolo di funzioni troppo complesse, sempre per riuscire a soddisfare il vincolo di universalità; se l'algoritmo richiede troppe risorse di calcolo diventa meno appetibile per dispositivi mobili o small-size di basse prestazioni e i tempi di attesa che ne risulterano renderebbero l'esperienza utente meno gradevole.

X Come ultimo punto citiamo ciò che non è una feature ma un set di tecniche usate nel riconoscimento dell'autenticità delle firme, che si è deciso di non includere nel progetto: la normalizzazione dei disegni, intesi come il riportare disegni simili alla stessa posizione, rotazione e dimensione. In [5] queste tecniche vengono applicate ai campioni delle firme, così da rendere più facile la comparazione e indipendente da differenti posizionamenti,rotazioni o dimensioni.
Nel caso del nostro progetto invece, visto che l'obiettivo non è limitato al solo confronto tra uguaglianza di figure, si è deciso di non implementare queste possibilità, per poter sfruttare la differenziazione tra utenti; come verrà spiegato nel paragrafo 3.1., conservare le differenze di posizione, rotazione e dimensione permette di differenziare, oltre che gli utenti uno dall'altro, anche gli eventuali impostori aumentando così la scalabilità e la sicurezza.

2.6. Tecniche algoritmiche scartate: motivazioni

Questo paragrafo riassume alcune tecniche tentate durante lo svolgimento del progetto ma poi tralasciate perché ritenute poco promettenti. Questa evenienza si è verificata principalmente in due occasioni.

2.6.1. Calcolo della "differenza" tra disegni in modo non lineare

La funzione che calcola la differenza tra due realizzazioni di un disegno nell'attuale progetto, per quanto riguarda le feature coordinata X, Y e pressione della penna, funziona calcolando (dopo varie operazioni di pre-processing descritte nella terza parte del testo) la somma S degli

scostamenti punto per punto nei vettori che ne rappresentano l'andamento temporale,secondo la formula :

$$S = \sum_{i=1}^{350} |X(i) - E_n(i)|$$

Dove X ed E_n sono i due vettori sotto confronto. Per maggiori dettagli su questa operazione, si veda il paragrafo 3.3.

Si è pensato, durante la creazione del sistema, di calcolare in modo alternativo lo scostamento tra due vettori e cioè, per ogni punto, di elevare al quadrato la differenza prima di sommarla al contributo degli altri punti anziché sommarla semplicemente, secondo la formula:

$$S = \sum_{i=1}^{350} (X(i) - E_n(i))^2$$

Lineare	Quad	Correttezza
11	1,2	si
16	1,84	si
8,1	1	si
16,08	2	si
9,1	1,1	si
20,2	2,78	si
18	2,1	si
30	4,4	si
31	5	no
22,8	3	si
37,7	6,5	no
20	2,7	si
15	1,63	si
42	9	no
45	11	no
11	1,44	si
64	17	no
40	10	no
36	8	no
52	14	no

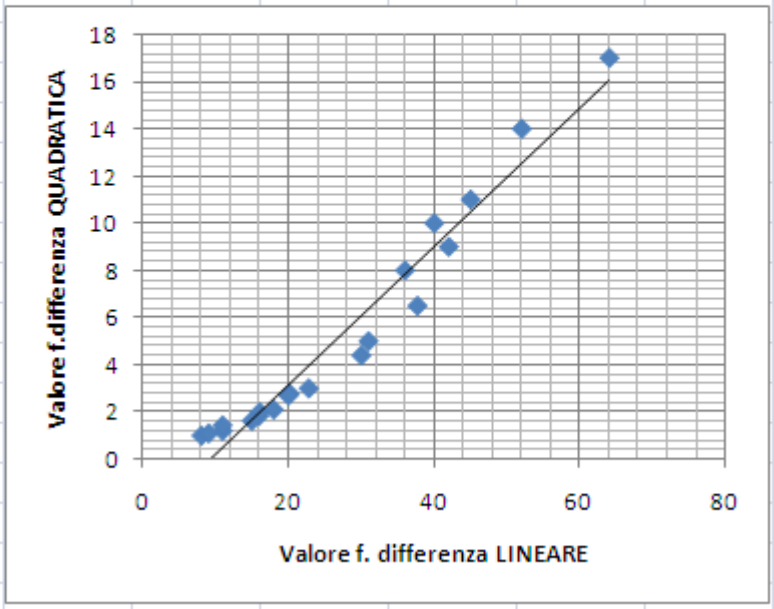

Figura 6 - Valori di "differenza" tra figure chiave. La prima colonna mostra i valori (numeri puri) ottenuti se calcolati in modo lineare (si noti che più il valore è alto più la somiglianza è cattiva; la soglia tra accettabilità e non accettabilità è fissata in circa 30); la seconda colonna mostra i valori ottenuti dal calcolo in modo quadratico, la terza colonna indica se ciascun disegno era corretto e quindi doveva essere accettato oppure no. Dal grafico si nota che i punti invece di distribuirsi su una curva netta si avvicinano abbastanza alla linea di tendenza retta, vanificando il vantaggio teorico dell'utilizzo della funzione quadratica.

Visto che la funzione differenza restituisce un numero che aumenta all'aumentare della diversità fra i due disegni, questo accorgimento avrebbe permesso di attribuire un peso inferiore alle differenze di piccola entità, e un peso superiore alle differenze più marcate; questo sulla base del fatto che piccoli numeri elevati al quadrato crescono meno di grandi numeri elevati al quadrato.

Ebbene, questo trucco molto promettente sulla carta, non ha dato i risultati sperati: dopo svariati test di disegno infatti, si è visto che i valori di differenza calcolati in entrambi i modi crescevano pressoché allo stesso modo; in Figura 6 si ha un plot di alcuni valori nel foglio elettronico, e come si vede, l'andamento esponenziale si nota appena e non è regolare nei valori di "fondo scala" (corrispondenti ad esecuzioni del disegno estremamente diverse da quelle originali). Si è poi ripetuto il test calcolando il valore come somma dei valori assoluti delle differenze elevate al cubo, e gli esiti, visibili in Figura 7 sono stati anch'essi poco promettenti. Questi risultati hanno così spinto a tralasciare la modifica e a continuare ad usare la funzione differenza lineare, perché più regolare e perché, nella fase di costruzione del progetto, anche più intuitiva per il debug.

Lineare	Cubica	Correttezza
12,5	0,88	si
12	0,4	si
41	3,06	no
33	2,4	no
44	3,28	no
22,2	0,8	si
33	1,83	no
37	2,3	no
11,6	0,4	si
8	0,4	si
30	2	no
33	2,06	no
137	36	no
98	17	no
87	18	no
70	9	no
121	26	no

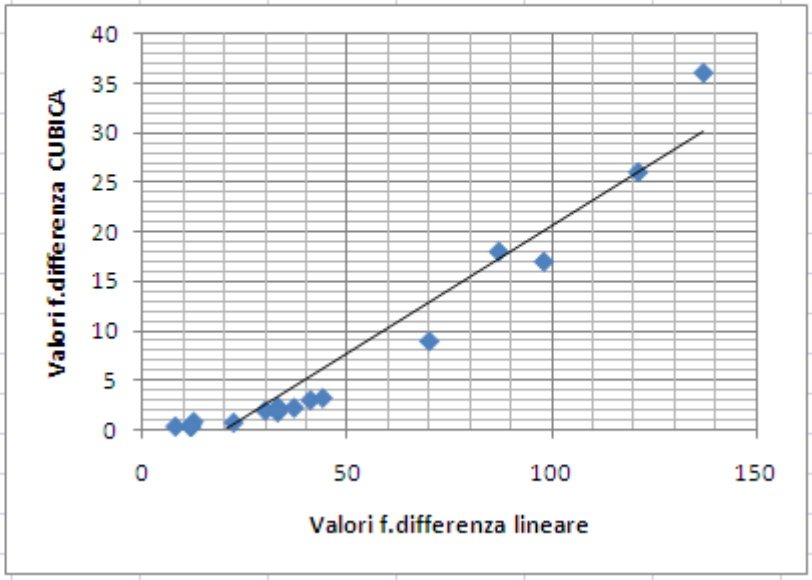

Figura 7 – Analogamente alla figura precedente, stavolta i dati si riferiscono al confronto tra funzione lineare e funzione cubica. Anche qui la curva è appena percettibile e sostanzialmente simile alla retta.

2.6.2. Uso del classificatore SVM per la verifica dell'utente

Questo tentativo è stato fatto nella parte iniziale del progetto; prima di partire con lo sviluppo dell'algoritmo ora utilizzato infatti, si era provato ad utilizzare una Support Vector Machine (SVM)

[19] (cioè un classificatore automatico) per determinare, caso per caso, se il disegno fosse stato fatto da un utente autorizzato oppure no.
Senza entrare nei dettagli della teoria dei classificatori (per i quali si rimanda alle trattazioni dedicate o a [19]), diciamo che una SVM è una macchina che, come ogni classificatore, ha bisogno di una fase di training; nella fase di training si fornisce una serie di vettori di valori numerici, in cui ogni vettore è contrassegnato dalla categoria a cui appartiene, e quando il numero di tali vettori campione è sufficientemente grande e significativo, la macchina riesce ad apprendere come classificare altri vettori della stessa natura ma con contenuto diverso da quelli usati nel training. Applicato al nostro caso, per ciascun utente, prendendo in input i dati grezzi di alcuni disegni "autorizzati" e alcuni disegni "non autorizzati", dopo la fase di training riuscirebbe a distinguere automaticamente tutti i disegni fatti in fase di Login stabilendo se appartengono all'utente dichiarato oppure no. Con "dati grezzi", si intende semplicemente la successione dei vettori delle coordinate X, Y e del vettore di pressione ottenuti da un generico disegno e normalizzati per essere ricondotti tutti alla stessa lunghezza.

Si è così cercato un classificatore SVM libero, interpellabile da linea comando quale LIBSVM [20] e una volta compreso il funzionamento di base si è provato a effettuare un training per ogni utente del sistema che riconoscesse se per tale utente i disegni chiave erano autorizzati o meno.

Per raggiungere questo scopo, venivano forniti al classificatore vettori di 3 categorie di disegni :

- Categoria 1: vettori di dati grezzi ottenuti da disegni di figure autorizzate all'accesso per l'utente in considerazione (ad esempio una casetta stilizzata);
- Categoria 2: vettori di dati grezzi di figure chiave diverse da quella scelta dall'utente (quindi secondo l'esempio precedente campioni di tutte le figure predefinite eccetto le casette)
- Categoria 3: vettori di dati grezzi di figure chiave uguali a quella scelta dall'utente (es. la casetta) ma disegnate da altre persone.

Nei primi test effettuati, sono state date in pasto al classificatore solo le categorie 1 e 2 : cioè è stato fatto il training solo con le casette e le non casette. I primi tentativi, effettuati con i parametri predefiniti di LIBSVM [20] , sono stati sconfortanti: la "accuracy" (cioè la percentuale di istanze riconosciute correttamente) si aggirava attorno al 45% , cioè meno di metà dei casi venivano indovinati, alla pari quindi di tirare a caso.
Il passo successivo è stato ovviamente quello di documentarsi sui parametri disponibili per LIBSVM e su come era possibile personalizzarli; in [20] si trovano guide e spiegazioni a riguardo.
E' bastato poco sforzo per scoprire che cambiando la modalità del classificatore in "nu-SVC" ([21] e [22]) l'"accuracy" passava al 100%: fornendo solo 15-20 vettori alla fase di training, il classificatore era in grado di indovinare col 100% di accuracy sequenze di 10-15 vettori alla volta. Risultato estremamente motivante, ma non sufficiente: infatti si stava ancora lavorando con le sole categorie 1 e 2, quindi il sistema era ancora impossibilitato a riconoscere una "casetta autorizzata" da una non autorizzata.

Si è allora aggiunta la terza categoria delle casette non autorizzate; l'accuracy è così scesa attorno all'80%; stesso risultato fondendo la categoria 3 con la categoria 2.

Questo peggior risultato è comprensibile e motivato dal fatto che la differenza tra le casette autorizzate e quelle non autorizzate è abbastanza ridotta rispetto alla differenza tra una casetta e una non casetta.
La ridotta differenza tra categoria 1 e categoria 3 è addirittura nulla se pensiamo al fatto che la SVM ottiene dati grezzi relativi solo alle coordinate X e Y e alla pressione; di conseguenza era sufficiente che due persone disegnassero la stessa cosa con pressione simile perché il classificatore non fosse più in grado di distinguerle.

Da questo momento si è resa ancora più evidente la necessità di utilizzare anche i dati provenienti dalle altre feature considerate, come tempo, numero di punti di attacco e loro posizione; ma siccome queste feature fornivano dei singoli numeri interi, anziché fornire un vettore, includere questi dati grezzi nell'input del classificatore sarebbe stato poco utile perché una decina di valori nulla potevano contro 350*3=1050 valori costituenti i 3 vettori X,Y e pressione già citati, se non con un numero molto elevato di campioni di training.

A questo punto l'unica misura adottabile per continuare sulla strada dell'SVM sarebbe stata quella di fornire un solido training capace di far distinguere le due categorie al classificatore; ma un training sufficientemente solido richiedeva qualche centinaia di campioni per ciascun tipo di disegno, che non c'era modo di ottenere in tempi brevi perché richiedeva di coinvolgere almeno 5-10 persone per una quantità non indifferente di tempo; inoltre, ipotizzando che ciò fosse stato possibile, ed ipotizzando che il risultato fosse stato positivo riportando l'accuracy a livelli di almeno il 90-95%, sarebbero comunque rimaste alcuni problemi ed alcuni lievi svantaggi nel sistema così progettato, riassunte nel seguito:

- Sarebbe stato impossibile permettere agli utenti del sistema di effettuare delle sigle o dei disegni personalizzati; questo perché se un utente si inventa un disegno non previsto tra quelli predefiniti, nel database della totalità dei disegni utilizzabili per il training non ci sono campioni per la categoria 3, cioè "disegni uguali a quelli fatti dall'utente ma realizzati da altre persone". Questo implica che il classificatore non potrebbe allenarsi su tale categoria e avrebbe quindi delle difficoltà a distinguere disegni realizzati dall'effettivo utente autorizzato da quelli di eventuali impostori, nuocendo gravemente alla sicurezza del sistema.
- La mancanza di figure personalizzate obbliga ovviamente a poter utilizzare solo quelle predefinite suggerite dal sistema, dando luogo ad un altro problema:
    - Se le figure predefinite sono poche succede che all'aumentare degli utenti scende ancora la sicurezza,perché se molti utenti utilizzano la stessa figura chiave per entrare nel sistema aumenta la probabilità che due utenti facciano il disegno con caratteristiche somiglianti, o perlomeno, giudicate somiglianti dal classificatore;

    - Se le figure predefinite sono molte, oltre ad essere necessario un significativo sforzo del progettista, si ha il problema che per raccogliere i campioni da inserire nella categoria 3 bisognerebbe far disegnare ad ogni utente in fase di Enrollment, se non tutte (in caso gli utenti siano pochi) almeno una parte delle figure a disposizione, rendendo così la procedura di Enrollment più lenta, complicata e meno user-friendly.
- Fasi di training con così tanti campioni avrebbero appesantito l'algoritmo, richiedendo più tempo e più risorse di calcolo, comportando problemi e rallentamenti se utilizzato su dispositivi mobili e andando in contrasto con l'obiettivo di universalità (per maggiori dettagli si vedano i paragrafi 0.1. e 4.1.1).

Considerati tutti questi aspetti quindi, si è deciso di accantonare l'uso della SVM per la verifica dell'identità. Questo è in parte giustificato dal fatto che quanto svolto è più adatto a situazioni con pochi utenti, e con figure chiave utilizzate da al più un solo individuo; questi meccanismi infatti erano stati tratti dal paper sul riconoscimento dell'identità della firma [5], dove una tal firma è utilizzata da un solo utente e le firme confrontate sono relativamente poche.

Visto quindi che le nostre necessità e i nostri obiettivi erano diversi e più vasti, si è passati al progettare da zero un algoritmo differente, basato su confronti anziché sull'utilizzo di classificatori, che è quello che è poi gradualmente cresciuto fino a diventare l'attuale algoritmo del sistema funzionante.

Con ciò non si vuol dire che la SVM sia una tecnica da scartare a priori per un progetto del genere; ma solo che in mancanza di solide quantità di campioni di training non è possibile procedere nell'approfondimento di quello che, perché no, potrebbe diventare un altro valido algoritmo. Potrebbe questo essere uno dei "to-do" per eventuali future espansioni del progetto.

3. Parte terza: Funzionamento ed uso del sistema

3.1. Vincoli da rispettare per l'utilizzo del sistema

Come in ogni gioco, anche se questo non è un gioco, ci sono delle regole da rispettare. Premesso innanzitutto che il sistema, così com'è, è sviluppato per essere in grado di prendere una decisione il più precisa possibile imponendo il minor numero possibile di vincoli, ci sono ovviamente delle regole da rispettare per ottenere dei buoni risultati.

Intanto iniziamo col dire che ogni utente di questo sistema dovrebbe aver preso un po' di familiarità con l'hardware che si troverà ad utilizzare (ad esempio la penna del Tablet PC) per usufruirne. Diversamente sarebbe come chiedere ad un bambino che non sa correre in bicicletta di salirci sopra e di correre subito senza mani sul manubrio: un po' troppo prematuro. Una volta che l'utente ha preso mano con l'hardware e sa ripetere dei disegni (o delle sigle o quant'altro) in modo abbastanza simile nelle varie ripetizioni, ed è pronto per la fase di Enrollment, deve tenere presente quanto segue:

- Le varie ripetizioni del disegno scelto come figura chiave, sia in fase di Enrollment che in fase di Login per poter essere riconosciuti, devono essere ripetute in modo sufficientemente simile, il che si traduce nel disegnare le figure in modo che ci sia una buona somiglianza nella posizione occupata nel quadro di lavoro, nella sua dimensione, e nella pressione impiegata alle varie ripetizioni. Questo vincolo non deve spaventare, in quanto non si pretende una perfetta uguaglianza nei disegni (anzi, è bene che ci sia un po' di variabilità in fase di Enrollment, così il relativo algoritmo è in grado di calcolare correttamente le tolleranze che utilizzerà per prendere le decisioni in fase di Login) , ma serve solo per ricordare che se ad ogni ripetizione del disegno si cambiano radicalmente le suddette caratteristiche, il sistema non sarà in grado di offrire risultati soddisfacenti. Inoltre rispettare questo accorgimento è un eccellente metodo per aumentare la sicurezza nell'accesso; un utente malintenzionato, anche se fosse venuto a sapere qual è la figura chiave da disegnare per entrare nel sistema, è fortemente improbabile che indovini contemporaneamente la posizione, la geometria, la rotazione, la pressione e la velocità di disegno visto l'elevatissimo numero di combinazioni possibili.
  In Figura 8 è mostrato un tris di ripetizioni corrette, mentre in Figura 9 si vede un tris di ripetizioni errate perché differenti in dimensione e posizione.

Figura 8- Esempio di come disegnare correttamente tre ripetizioni di una stessa figura chiave, per garantire una buona qualità di riconoscimento

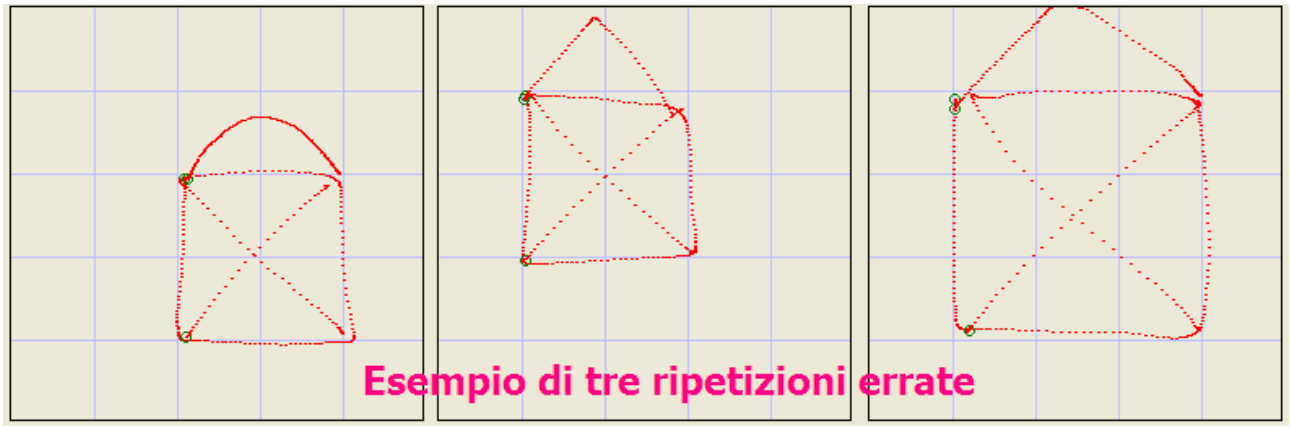

Figura 9 - Esempio di tre ripetizioni disegnate in modo sconveniente, che sono rifiutate dal sistema perché abbasserebbero la qualità di riconoscimento ed innalzererebbero il rischio di accesso da parte di impostori

- Nel comporre le varie ripetizioni del disegno bisogna eseguire i tratti sempre nello stesso ordine e direzione. In altre parole, se il mio disegno è una casetta, non posso partire una volta dalla base e un'altra volta dal tetto; per ovviare a questo problema è sufficiente fare qualche esercizio prima della fase di Enrollment, in cui si decide come disegnare la propria figura chiave e lo si tiene a mente per il futuro. Come il vincolo precedente, anche questo aiuta significativamente la sicurezza.
- Terzo vincolo, derivante dal precedente ma forse meno intuitivo, è quello che richiede di mantenere nella composizione del disegno gli stessi punti di attacco (cioè i punti in cui appoggio la penna sullo schermo per iniziare i tratti).
  Per fare un esempio, se la figura chiave è un triangolo isoscele e si è deciso di disegnarlo iniziando dalla punta in alto e completando la figura con una sola spezzata staccando la penna dallo schermo solo quando si è chiuso il triangolo, alle successive ripetizioni si dovrà disegnarlo sempre così; se invece si stacca la penna dopo ogni lato, il sistema lo considera come un disegno diverso, anche se si è proceduto con lo stesso ordine di tratti (in caso di errore in fase di Enrollment viene chiesto di rifare il disegno e in fase di Login viene rifiutato l'accesso).
  Questo avviene allo scopo di aumentare le possibilità di disegno di una stessa figura, per

cui essendoci più modi di disegnare una stessa figura aumenta la sicurezza ed è più facile distinguere l'utente autorizzato da un impostore. Inoltre, se si è sbagliato a disegnare un tratto o una parte di disegno, mai tornare indietro per ripassarci sopra. In tal caso si deve continuare fino al termine del disegno oppure cancellarlo e ripeterlo.

- Ogni disegno, sia in fase di Enrollment che di Login, deve essere completato in un tempo inferiore ai 7 secondi, ritenuto in fase di progettazione più che sufficiente per realizzare un disegnino stilizzato od una sigla. Tale vincolo temporale è stato posto per enfatizzare il concetto che i disegni in questa sede non devono essere realizzati con l'obiettivo di essere fatti "bene" , ma bensì "velocemente" come una sigla o una firma. L'esercitarsi a fare disegni stilizzati velocemente, permette una certa variabilità (infatti le ripetizioni dei disegni non devono mai essere assolutamente identiche tra loro, altrimenti i confronti diventano troppo stringenti e diventa difficile essere riconosciuti correttamente), ed inoltre permette di "sfoggiare" meglio al sistema qual è il proprio comportamento nei termini delle feature analizzate (si pensi a titolo di esempio al fatto che dei disegni fatti troppo lentamente, con l'intento di essere precisi, differiscono molto in termini di tempo, pressione, numero di tratti rispetto ad una firma che invece con l'abitudine viene sempre abbastanza simile).
- Merita infine di essere citato quello che non è un vero vincolo ma è un utile consiglio per ottenere buone prestazioni: realizzare il disegno impiegando per quanto possibile gli stessi tempi e la stessa pressione su schermo, in modo da rendere più facile distinguere l'utente autorizzato dagli impostori.

Al momento del disegno di una figura, la pressione sullo schermo, a differenza di dimensione e posizione del disegno che si decidono facilmente, è un po' meno facile da controllare perché dipende da vari fattori, come la posizione della mano al momento del disegno, l'orientamento dello schermo in caso di tablet PC, la stanchezza, il livello di stress, eventuali tensioni emotive immediatamente precedenti il disegno, possibili dolori o infortuni alla mano ed altri innumerevoli fattori psico-motori.

Questo è uno dei motivi per cui al termine della fase di Enrollment appare un riepilogo che mostra la bontà dei tre fattori di variabilità riscontrati durante le 10 esecuzioni, seguiti da una valutazione della bontà delle stesse. Qualora una o più di tali variabilità non risultassero ad un livello buono (contrassegnato da un colore verde), è consigliabile ripetere la procedura di Enrollment, eventualmente in un momento di maggior tranquillità; questo perché se le variabilità riscontrate sono troppo elevate (e quindi giudicate come Discreta o perfino Scarsa), il sistema al momento del Login sarà troppo tollerante e si rischia che qualche persona non autorizzata riesca ad accedere al posto del legittimo proprietario. Per maggiori dettagli sulla fase di Enrollment, le variabilità e la regolazione anche manuale delle rispettive tolleranze, si rimanda ai paragrafi dedicati.

3.2. L'algoritmo di Enrollment

Il sistema è composto sostanzialmente da due parti distinte, la procedura di Enrollment e la procedura di Login, come già detto nel paragrafo 2.2. Verranno ora descritte entrambe le procedure, parlando sia dell'algoritmo sottostante, sia dei passi necessari per l'uso della procedura (una sorta di guida all'uso).

In questo paragrafo verrà descritto il funzionamento dell'algoritmo di Enrollment, cioè la fase in cui vengono raccolti per la prima volta i dati biometrici di un certo utente per imparare a verificare la sua identità.

La procedura inizia con una maschera come quella di Figura 10, che chiede all'utente il suo nome e la figura che intenderà utilizzare come figura chiave; potrà infatti scegliere tra 7 figure predefinite, (quelle proposte coniugano la caratteristica di semplicità con la possibilità di disegnarle in parecchi modi distinti) e una figura a piacere, permettendo così all'utente di inventare qualcosa di sua fantasia ma anche di utilizzare qualcosa che è già abituato a fare spesso in modo meccanico come una sigla o le proprie iniziali. Si nota che la possibilità di utilizzare sia figure chiave predefinite sia personalizzate aumenta il livello di sicurezza, in quanto il numero di figure chiave possibili diventa potenzialmente infinito.

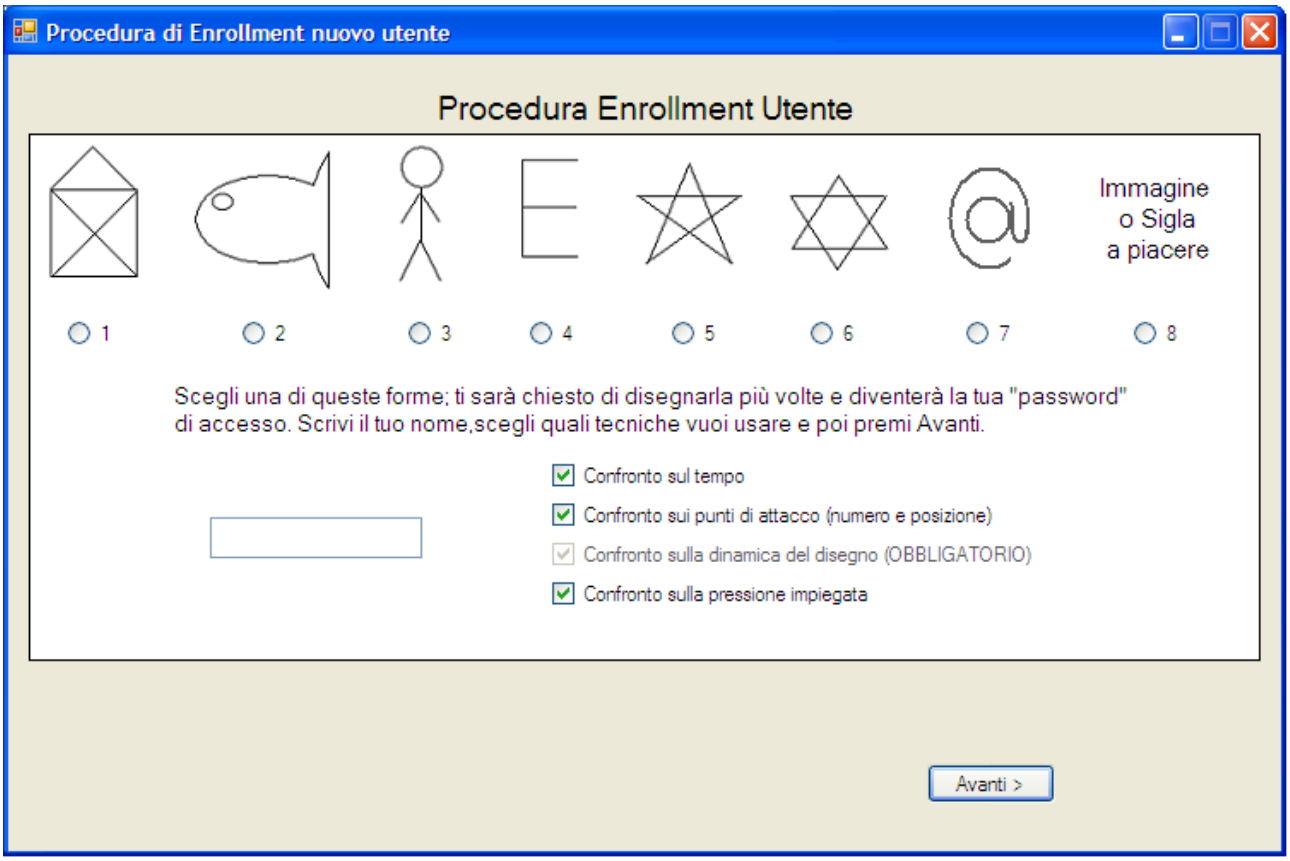

Figura 10 - Prima fase della procedura di Enrollment: scelta nome, figura chiave, personalizzazione feature

Una volta scritto il proprio nome e scelta la figura chiave desiderata, prima di continuare è possibile (ma facoltativo) personalizzare quali feature di valutazione si vogliono osservare in fase di Enrollment. L'impostazione predefinita, che è anche quella che offre la massima sicurezza, seleziona tutte le feature; in caso si decida di disattivarne alcune, non saranno analizzate in fase di Enrollment e di conseguenza non saranno utilizzabili per i confronti in fase di Login. Per maggiori dettagli su questo aspetto si rimanda alla trattazione dettagliata del paragrafo 3.4., mentre per la descrizione completa delle feature stesse si veda il precedente paragrafo 2.4.

Successivamente viene chiesto di disegnare per 10 volte la propria figura chiave; per far bene questi 10 disegni, e quindi per ottenere dei buoni risultati/performance al momento del Login, è necessario rispettare i vincoli enunciati nel paragrafo 3.1. cioè, dopo aver preso un po' di confidenza con la penna/tavoletta, ripetere i disegni in modo simile sotto gli aspetti di geometria, ordine dei tratti, tempo impiegato e pressione.

Durante queste 10 ripetizioni del disegno, come si nota dalla Figura 11, il sistema controlla in tempo reale che sia rispettato il più stringente dei vincoli cioè quello sul numero di punti di attacco della penna impiegati per il disegno della figura; la scritta rossa in basso a destra mostra, dopo il primo disegno, quanti punti di attacco sono stati impiegati e richiede che questo numero sia lo stesso per tutte le successive esecuzioni. Nel caso questo numero differisca in qualcuna delle ripetizioni, magari per una distrazione o per motivi di scarsa pratica con la penna, un messaggio avvisa che è necessario ridisegnare la figura per renderla coerente con le precedenti.

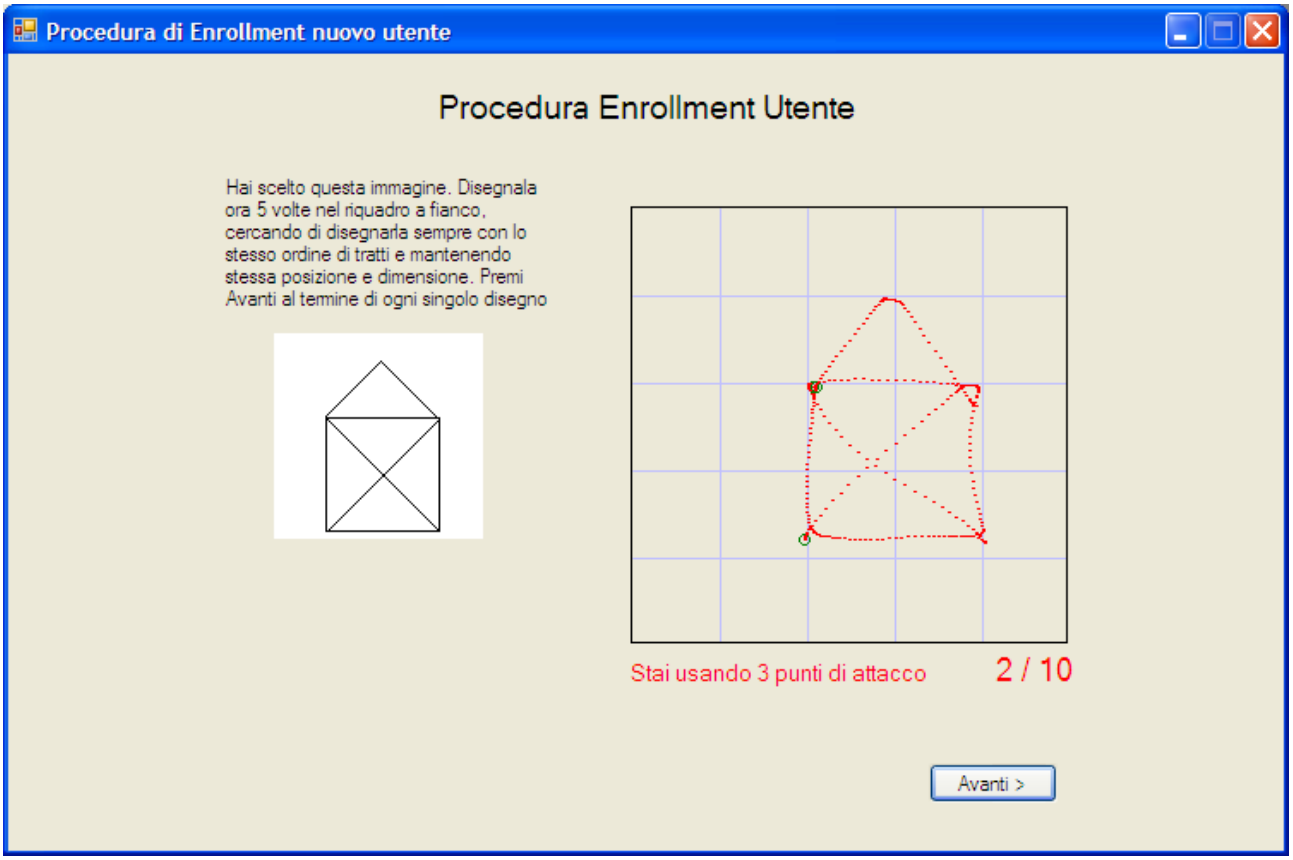

Figura 11 - Seconda fase della procedura di Enrollment : disegnare più volte la figura chiave scelta

Una volta terminati i 10 disegni, l'algoritmo si occupa di fare le seguenti elaborazioni:

- ***Normalizzare*** i vettori e salvarli su file: I vettori contenenti l'andamento temporale della pressione e delle coordinate X e Y vengono normalizzati portando il numero di campioni dal valore N (variabile ad ogni esecuzione e pari al valore T_i del tempo impiegato) ad una dimensione fissa di 350 punti (ritenuta sufficiente per rappresentare con buon grado di dettaglio qualsiasi figura schematica si sia disegnata). Normalizzare il vettore in questo modo comporta un ricampionamento del valore di ogni singolo punto, il quale si effettua tramite interpolazione se N < 350 oppure tramite decimazione se N > 350 (Algoritmo spiegato nel paragrafo 3.3)
- Calcolare la massima ***variabilità sul tempo di esecuzione*** ricavato dalle 10 acquisizioni: Per calcolare questo valore, dati i dieci tempi $T_1..T_{10}$ propri di ciascuna esecuzione, si calcola per prima cosa la media pesata T_{med} :

$$T_{med} = \frac{\sum_{i=1}^{10}(T_i * P_i)}{\sum_{i=1}^{10} P_i}$$

dove i pesi $P_1..P_{10}$ sono dei valori interi nell'intervallo [1,3]; se $P_i = 1$ significa che il dato ottenuto dalla i-esima ripetizione del disegno deve avere influenza minima sul calcolo della media, viceversa quando $P_i=3$ l'influenza sarà massima. L'attuale vettore dei pesi è settato come segue:

$$P = \{1,2,3,3,3,3,3,3,3,2\}$$

valori motivati dal fatto che nei primi 2 disegni l'utente potrebbe non aver ancora preso bene la mano con la penna o con la gestualità necessaria alla realizzazione della figura chiave, e nell'ultimo disegno invece, consapevole di aver quasi finito, potrebbe realizzarlo in modo affrettato o impreciso. Ottenuto il valore T_{med}, si calcola ora il massimo scostamento T_s tra T_{med} e i 10 tempi ottenuti dalle esecuzioni mediante la formula :

$$T_s = \max\left(T_{med} - T_{min}\,, T_{max} - T_{med}\right)$$

dove T_{min} e T_{max} rappresentano rispettivamente i valori del più basso e del più alto tempo impiegato nelle 10 esecuzioni.
Il valore Ts così calcolato assume il nome di ***variabilità***, che verrà salvato sul file dei dati dell'utente sotto Enrollment e che verrà mostrato nella pagina di riepilogo al termine della procedura. Lo scopo di tale schermata verrà discusso nel seguito del paragrafo.

- Calcolare la massima ***variabilità sulla posizione dei punti di attacco***: siccome i vincoli di utilizzo comportano che la figura chiave deve essere disegnata con lo stesso ordine di tratti e approssimativamente nella stessa posizione, si ha che i punti di attacco (rappresentati dai cerchietti verdi visibili in Figura 11) dovrebbero trovarsi in posizione pressoché simile in tutte le ripetizioni. Il sistema registra al massimo 10 punti di attacco a_n, in quanto giudicati più che sufficienti per la realizzazione di qualsiasi figura semplice o sigla; nel caso l'utente generasse più di 10 punti di attacco, quelli eccedenti verranno ignorati.
  Il calcolo della variabilità dei punti di attacco viene effettuato in modo analogo a quello dei tempi di esecuzione: Per ogni punto di attacco a_n (con 1 <= n <= 10), si calcolano le medie pesate $X_{n,med}$ e $Y_{n,med}$:

$$X_{n,med} = \frac{\sum_{i=1}^{10}(X_{n,i} * P_i)}{\sum_{i=1}^{10} P_i} \qquad Y_{n,med} = \frac{\sum_{i=1}^{10}(Y_{n,i} * P_i)}{\sum_{i=1}^{10} P_i}$$

  dove $X_{n,i}$ rappresenta il valore della coordinata X dell'n-esimo punto di attacco all'i-esima iterazione, ed analogamente $Y_{n,i}$ per l'asse Y.Ora si calcola la variabilità A_s ottenuta come massimo scostamento esistente, su ambo le coordinate, tra la media e i valori di coordinata dei punti.

$$A_s = \max_{\forall n} \left(\max \left(X_{n,med} - X_{n,min}, X_{n,max} - X_{n,med}, Y_{n,med} - Y_{n,min}, Y_{n,max} - Y_{n,med}\right)\right)$$

  Dove $X_{n,min}$, $X_{n,max}$, $Y_{n,min}$ e $Y_{n,max}$ sono le coordinate minime e massime ottenute per l'n-esimo punto di attacco. L'unificazione dei dati delle due coordinate è fatta sulla base del pensiero che se un utente varia di una certa quantità su uno dei due assi è probabile che lo faccia anche sull'altro. Questo valore As assume il nome di ***variabilità sui punti di attacco***,verrà anch'esso salvato su file e mostrato nella schermata di riepilogo.

- Calcolare la massima ***variabilità sulla pressione della penna*** sullo schermo: Per fare questo, si procede prima ad una operazione di "stretch" sui vettori registranti i dati della pressione (si veda il funzionamento di questa operazione nel successivo paragrafo) su tutte le 10 ripetizioni, per ottenere dei dati allineati e quindi confrontabili tra loro. Dopodichè, per ogni i-esimo punto del vettore (con 1 <= i <= 350, come vedremo nel seguito), viene calcolata la differenza tra il massimo e il minimo trovati nel punto i delle 10 ripetizioni. I valori assoluti di queste differenze vengono sommati e la somma costituisce la variabilità massima.

I tre valori di variabilità annunciati poc'anzi vengono salvati nel file che contiene i dati relativi all'utente e vengono poi utilizzati per il calcolo delle tolleranze al momento del Login. Per vedere meglio come queste variabilità influiscono si rimanda ai paragrafi dedicati 3.3. e 3.4.

Il numero di punti di attacco (che è l'unica feature non ancora citata) essendo fisso e non richiedendo elaborazioni viene salvato nel file così com'è.

Prima di concludere il paragrafo, ci si vuole soffermare sull'ultima schermata dalla procedura di Enrollment che mostra un riepilogo come quello della figura Figura 12. Esso riporta i valori numerici delle 3 variabilità ottenute ed indica una stima della bontà delle stesse, espressa tramite una scala di tre giudizi: {Buono, Discreto, Scarso}. Il giudizio Buono si ottiene se la variabilità è inferiore ad un certo valore impostato come riferimento; Il giudizio Discreto si ottiene se la variabilità è compresa nell'intervallo tra il valore di riferimento ed il valore stesso moltiplicato per 1,4. Il giudizio Scarso si ottiene se la variabilità è maggiore di 1,4*Valore di riferimento. I valori di riferimento citati sono diversi per ognuna delle 3 feature, e sono stati stimati in fase di progettazione in seguito a numerose prove empiriche; il loro valore è pari a:

- 45 per la feature di tempo impiegato
- 20 per la feature di posizione dei punti di attacco
- 70 per la feature di pressione.

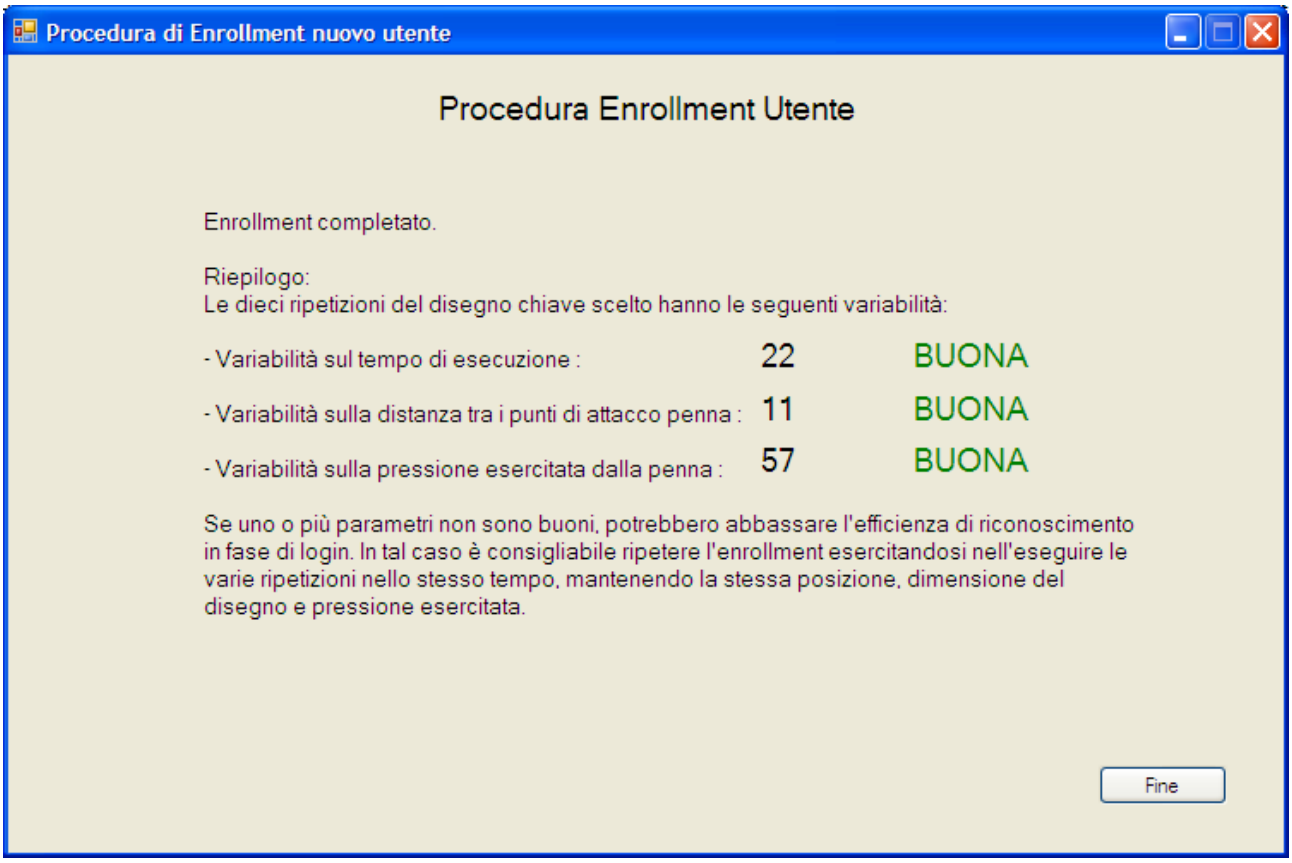

Figura 12 - Riepilogo della procedura di Enrollment: il giudizio sulla qualità dei dieci campioni del disegno.

Questo riepilogo è importante perché permette di capire se i dieci campioni della figura chiave sono sufficientemente simili da garantire buone prestazioni di riconoscimento anche in fase di Login. Se si ottengono dei valori di Discreto o Scarso su una o più feature infatti, significa che i campioni sono poco simili tra loro, che potrebbero provocare in fase di Login una diminuzione della sicurezza, in quanto con tolleranze più larghe il sistema diventa più permissivo e potrebbe facilitare l'accesso agli impostori. In caso si ottengano giudizi Discreti o Scarsi quindi, è consigliabile ripetere la procedura di Enrollment dopo essersi esercitati un po'.

3.3. L'algoritmo di Login

In questo paragrafo è descritto in modo dettagliato l'algoritmo di Login per accertare l'identità dell'utente. Questa volta, vista la maggior complessità dell'algoritmo, la descrizione del vero e proprio algoritmo precede ed è distinta dalle istruzioni per l'uso della relativa schermata.

Una volta ottenuto il nome dell'utente che vuole accedere al sistema e i dati del campione della figura chiave disegnata, l'algoritmo può iniziare.

Tale algoritmo si compone di 5 fasi, ripetute ciclicamente n volte dove n è il numero di campioni della figura chiave registrati nella procedura di Enrollment (cioè attualmente 10), dopodiché un'ultima operazione calcola un "punteggio di somiglianza" che permette al sistema di decidere se consentire l'accesso o meno. Si inizia con n=1:

- Prima fase: Test sul tempo impiegato. La prima fase effettua un confronto tra il tempo T_a impiegato a disegnare l'intera figura chiave nella esecuzione corrente e il tempo T_n impiegato nella n-esima realizzazione di Enrollment. Sia data la variabilità V ottenuta dalle elaborazioni post-Enrollment, si definisce la tolleranza Z = 2V. Se $T_n - Z < T_a < T_n + Z$ il test si ritiene superato e si passa alla fase successiva. Se in caso contrario si eccedono i limiti (la figura chiave è stata disegnata troppo velocemente o troppo lentamente) il test fallisce e l'algoritmo, senza eseguire le rimanenti 4 fasi, passa al ciclo successivo dove ricomincia i confronti tra la figura di Login e la (n+1)-esima figura di Enrollment.
  La tolleranza Z è definita pari a 2*V in seguito ad osservazioni empiriche: se definita pari a V infatti, il sistema diventa troppo severo. Se definita pari a P*V con P>2 invece, il sistema diventa troppo permissivo.

- Seconda fase : Test sul numero di punti di attacco.La seconda fase dell'algoritmo controlla che il disegno della figura chiave di Login abbia lo stesso numero N di punti di attacco utlizzati in fase di Enrollment. Se l'uguaglianza è verificata, il test si ritiene superato e si passa alla fase successiva. In caso contrario il test fallisce e l'algoritmo, senza eseguire le rimanenti 3 fasi, passa al ciclo successivo dove ricomincia i confronti tra la figura di Login e la (n+1)-esima figura di Enrollment.

- Terza fase : Test sulla posizione dei punti di attacco.Questa fase dell'algoritmo viene eseguita h volte, dove h è il numero di punti di attacco realizzati durante il disegno della figura chiave; sia data inoltre la variabilità V calcolata post-Enrollment, e si definisca la tolleranza Z = 2*V.Per ogni i-esimo punto di attacco, con 1 <= i <= h, si confrontano le coordinate X_i e Y_i rispettivamente con $X_{n,i}$ e $Y_{n,i}$ (cioè le coordinate dell'i-esimo punto nella n-esima realizzazione di Enrollment); se $X_{n,i} - Z < X_i < X_{n,i} + Z$ e $Y_{n,i} - Z < Y_i < Y_{n,i} + Z$, il test si ritiene superato e si passa alla fase successiva. In caso contrario il test fallisce per superamento della tolleranza e l'algoritmo, senza eseguire le rimanenti 2 fasi, passa al ciclo successivo dove ricomincia i confronti tra la figura di Login e la (n+1)-esima figura di Enrollment.
  La tolleranza Z è definita pari a 2*V in seguito ad osservazioni empiriche: se definita pari a V infatti, il sistema diventa troppo severo. Se definita pari a P*V con P>2 invece, il sistema diventa troppo permissivo.

- Quarta fase : Test sulle coordinate X e Y del disegno. Questa fase viene ripetuta per due volte, una prima volta lavorando sui dati relativi all'asse X del disegno, e poi (solo se si ha esito positivo) viene ripetuta una seconda volta sui dati relativi all'asse Y. Vista la perfetta uguaglianza delle due ripetizioni, nel seguito verrà descritta solo la prima.

    - Condizione iniziale: Come già precisato nel paragrafo 2.4., l'andamento nel tempo della coordinata X è contenuto in un vettore. Tale vettore, appena l'utente ha disegnato la sua figura chiave, ha una lunghezza variabile pari ad T, dove T è pari al numero di campioni ricevuti dall'hardware nel tempo impiegato alla realizzazione del disegno, contenuti nelle sue celle, senza scartarne alcuno; la penna (si veda il paragrafo 2.3.1.) infatti fornisce 133 campioni al secondo, di conseguenza T = 133*<tempo impiegato in secondi>. I valori contenuti nelle celle del vettore assumono valore reale nell'intervallo [0,1], dove 0 rappresenta l'estremo sinistro del riquadro di disegno ed 1 rappresenta l'estremo destro.
    - Passo 1 - ***Normalizzazione***: Questo vettore di lunghezza T viene ricondotto ad un vettore di lunghezza fissa, tale da facilitare le operazioni di confronto tra vettori. La taglia scelta per la lunghezza è pari a 350, cifra giudicata sufficiente per permettere un livello di dettaglio molto buono senza gravare troppo sulle prestazioni dell'algoritmo. Per passare da T a 350 campioni può essere necessaria una decimazione dei campioni (se N>350) oppure una interpolazione dei campioni (se N<350), obiettivo raggiunto tramite una funzione scritta ad hoc, visibile nel pseudocodice della Figura 13.

```
Normalizza( Old(), OldSize, NewSize )
        Dim New(NewSize), i, passo, prevI As Single
        Dim j As Integer
        passo = OldSize / NewSize
        If passo < 1 Then  'ALGORITMO DI INTERPOLAZIONE
            prevI = 0
            j = 1
            i = passo
            While j < NewSize
                If ⌈i⌉ > prevI Then
                    New(j) = Old(⌈i⌉)
                Else  'Interpola linearmente col punto successivo
                    New(j) = (Old(prevI) + Old(prevI + 1)) / 2
                End If
                prevI = ⌈i⌉
                j = j + 1
                i = i + passo
            End While
        Else                'ALGORITMO DI DECIMAZIONE
            j = 1
            i = passo
            While i <= OldSize
                New(j) = Old(⌈i⌉)
                j = j + 1
                i = i + passo
            End While
        End If
        Return New
```

Figura 13 - Pseudocodice algoritmo di Normalizzazione

- Passo 2 - ***Stretch***: Questo passo si applica solo se il disegno della figura chiave presenta 2 o più punti di attacco. Questa operazione detta di "stretch" effettua una nuova normalizzazione stavolta suddivisa in più zone, cioè tra ogni punto di attacco e il successivo, per fare in modo che nei vettori da confrontare i punti di attacco si trovino nella stessa posizione. Questo è necessario per evitare che la differenza di velocità tra le esecuzioni dei vari tratti interferisca nella valutazione della differenza nella traiettoria. Per fare un esempio esplicativo, se la mia figura chiave è composta da due soli tratti,ad una esecuzione potrei impiegare 0,5 secondi nel disegnare il

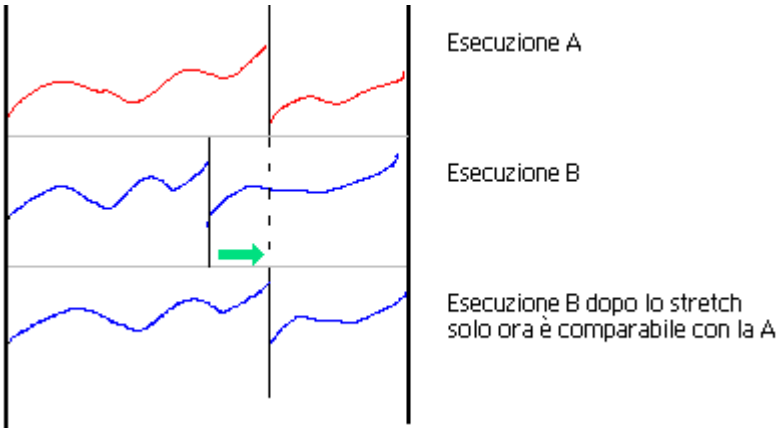

Figura 14 - L'algoritmo di stretch per migliorare il confronto tra disegni

primo tratto e 0,5 secondi nel disegnare il secondo, mentre ad una successiva esecuzione potrei impiegare rispettivamente 0,4 e 0,6 secondi. Il confronto dei vettori derivanti da queste due esecuzioni, senza applicare prima l'operazione di stretch, darebbe un risultato affetto da questo vizio di "umana irregolarità", che non avendo nulla a che fare con il confronto dell'effettiva traiettoria realizzata dalla penna, va eliminato a priori per non ottenere cattivi risultati. I grafici di Figura 14, in cui l'asse orizzontale rappresenta il tempo (espresso dai 350 campioni) e l'asse verticale il valore della coordinata, dovrebbero chiarire maggiormente il concetto. La Figura 15 mostra l'effettiva utilità dello stretch : La linea rossa corrisponde ad una figura chiave memorizzata in fase di Enrollment, che sta per essere confrontata con una figura chiave appena ricevuta dalla pagina di Login identificata dalla linea blu. Si nota che se si effettuasse una differenza tra la linea rossa e quella blu si otterrebbe un risultato falsato dalla non corrispondenza dei punti di attacco (indicati dai pallini rossi e blu). L'algoritmo di stretch allora processa il vettore corrispondente alla linea blu portando i punti di attacco a coincidere con quelli rossi ottenendo come risultato la linea verde. Il confronto tra la linea rossa e la linea verde è ora più significativo.
L'algoritmo che realizza l'operazione di stretch è del tutto analogo a quello che realizza la normalizzazione del passo 1, solamente eseguita su spezzoni del vettore anziché sull'intero vettore; si veda quindi ancora il pseudocodice di Figura 13 per una migliore comprensione.

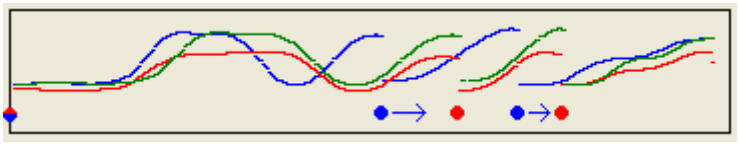

Figura 15 - Effetto dello stretch prima di un confronto tra due ripetizioni di una figura chiave

- Passo 3 - ***Confronto***: Ora si passa al vero e proprio confronto tra l'esecuzione corrente e quella fornita dai dati della n-esima realizzazione di Enrollment. Sia dato il vettore $E_n(i)$ contenente l'andamento temporale dell'asse X nella n-esima realizzazione di Enrollment, e sia X(i) il vettore contenente la stessa grandezza per la realizzazione di Login. Si calcola per prima cosa la somma S degli scostamenti definita come segue:

$$S = \sum_{i=1}^{350} |X(i) - E_n(i)|$$

A questo punto, se S < 33 (tolleranza fissa e ottenuta da analisi empiriche), il test è considerato superato e si passa ad effettuare le stesse operazioni sull'asse Y. In caso contrario il test fallisce e l'algoritmo, senza eseguire la rimanente fase, passa al ciclo successivo dove ricomincia i confronti tra la figura di Login e la (n+1)-esima figura di Enrollment.

- Quinta fase : Test sulla pressione esercitata dalla penna. Questa quinta ed ultima fase somiglia alla precedente, in quanto esegue gli stessi passi ma stavolta lavora, anziché sulle coordinate cartesiane del disegno, sui valori di pressione istantanea della penna sullo schermo, anch'essi come già detto registrati in un vettore. Anche qui dopo lo stretch si calcola la somma degli scostamenti S tramite la stessa formula; l'unica differenza rispetto alla fase precedente è il calcolo della tolleranza, che stavolta non è fissa ma è pari alla variabilità V calcolata post-Enrollment. Se S <= V il test si considera superato ed a questo punto,essendo l'ultimo test, viene registrato il successo del confronto tra il campione di Login e l'n-esimo campione di Enrollment. A questo punto se n < 10 lo si incrementa di una unità e si riparte dalla prima fase; se n = 10 si termina il ciclo.

Quando queste 5 fasi sono state ripetute per la decima volta, come già premesso, una volta per ogni campione di Enrollment,se il "punteggio di Somiglianza" , definito come il numero di campioni il cui confronto ha avuto successo diviso il numero di campioni totali) è maggiore o uguale a 7/10, si considera la procedura di Login conclusa con successo e quindi l'utente può avere accesso al sistema. In caso contrario l'accesso viene rifiutato senza dichiarare per quale motivo (in altre parole, senza specificare in quali fasi sono falliti i confronti) in modo che un eventuale aggressore non sia in grado di capire quali peculiarità del disegno dovrebbe migliorare, preservando così la sicurezza.

La Figura 16 mostra un possibile esito di questo algoritmo.

Figura 16 - Esempio di possibile esito dell'algoritmo di Login

L'algoritmo è a questo punto terminato.

Passiamo ora alla parte di "guida all'uso" per questa funzione. La pagina di Login si presenta come visibile in Figura 17. Per utilizzarla è necessario scrivere il proprio nome utente (lo stesso specificato in fase di Enrollment), disegnare la propria figura chiave nell'apposito riquadro e premere il tasto Entra.

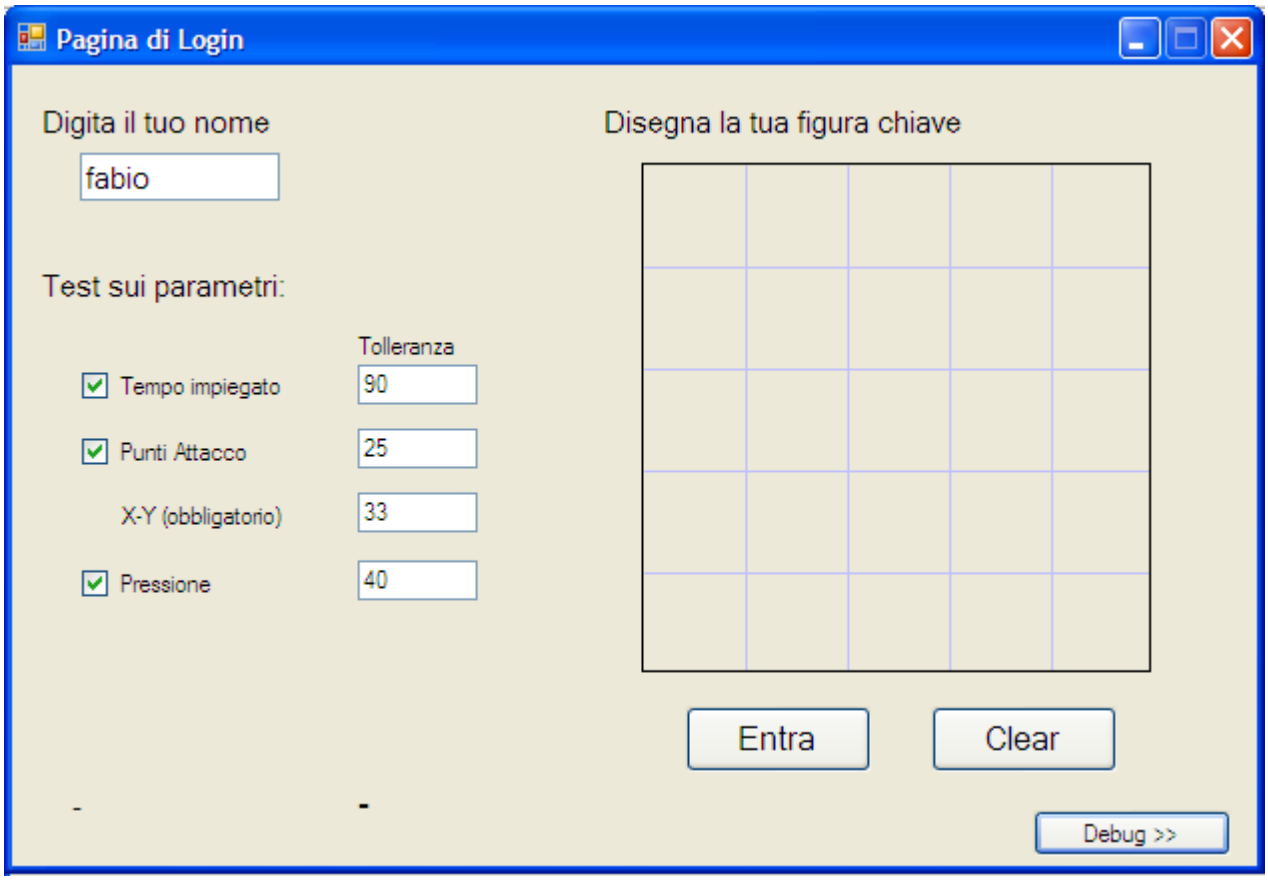

Figura 17 - Pagina di Login

Una volta scritto il proprio nome, la prima operazione che viene fatta è quella di caricare le tolleranze personalizzate per l'utente specificato e di mostrarle negli appositi riquadri; queste tolleranze sono direttamente proporzionali alle variabilità calcolate al termine dell'algoritmo di Enrollment (si veda il relativo paragrafo 3.2. per maggiori dettagli); esse infatti, se non vengono personalizzate, sono calcolate come segue:

- Per il tempo impiegato, la tolleranza Z è impostata al doppio della variabilità V ottenuta in fase di Enrollment. (Z = 2*V);
- Per la distanza sui punti di attacco, la tolleranza Z è anche stavolta impostata al doppio della variabilità V. (Z = 2*V);
- La tolleranza per la differenza riscontrata sugli assi X e Y è l'unica indipendente dall'utente ed impostata ad un valore pari a 33 (ricavata anch'essa in fase di progettazione in seguito a numerose prove empiriche).
- Per la pressione infine, la tolleranza Z è impostata allo valore della variabilità V ottenuta in fase di Enrollment, arrotondata all'intero superiore.

Tutte queste tolleranze, come visibile dalle caselle di testo editabili, possono essere variate a seconda delle necessità, sia nel valore sia nell'attivazione o meno dei confronti su tali feature; infatti, disattivando le relative spunte visibili in Figura 17 possono essere disabilitati i test sulle feature indicate nel caso che, tanto per fare un esempio, ci si trovi in situazioni in cui il tempo di esecuzione diventa molto variabile e non è possibile o consigliabile utilizzarlo come parametro di

confronto. Se alcune di queste feature sono già state disabilitate nella procedura di Enrollment, le relative spunte appaiono automaticamente oscurate e non selezionabili, proprio perché non esistono i dati per effettuare tali confronti.

Dopo aver caricato tali tolleranze, inizia l'algoritmo precedentemente descritto, ed il risultato è ottenuto sottoforma dell'indicazione del punteggio di somiglianza e di una dicitura "ACCETTATO" oppure "Rifiutato", come visibile rispettivamente in Figura 18 e Figura 19. Come si nota nella Figura 19, pur essendo la figura chiave della stessa natura e disegnata approssimativamente nello stesso tempo, con lo stesso ordine di tratti e con lo stesso numero di punti di attacco, viene rifiutata (con un punteggio somiglianza di 0/10), perché non è disegnata nella stessa posizione in cui il vero utente l'avrebbe disegnata, e quindi fallisce i test sulla posizione dei punti di attacco (e pure i test sulle coordinate X e Y se venissero effettuati, il che non accade perché il fallimento sulla posizione dei punti di attacco interrompe il ciclo di test). Inoltre si noti che non è necessario che la figura sia disegnata in modo perfetto, perché il sistema valuta non la regolarità dei tratti, ma il comportamento assunto dall'utente mentre lo disegnava.

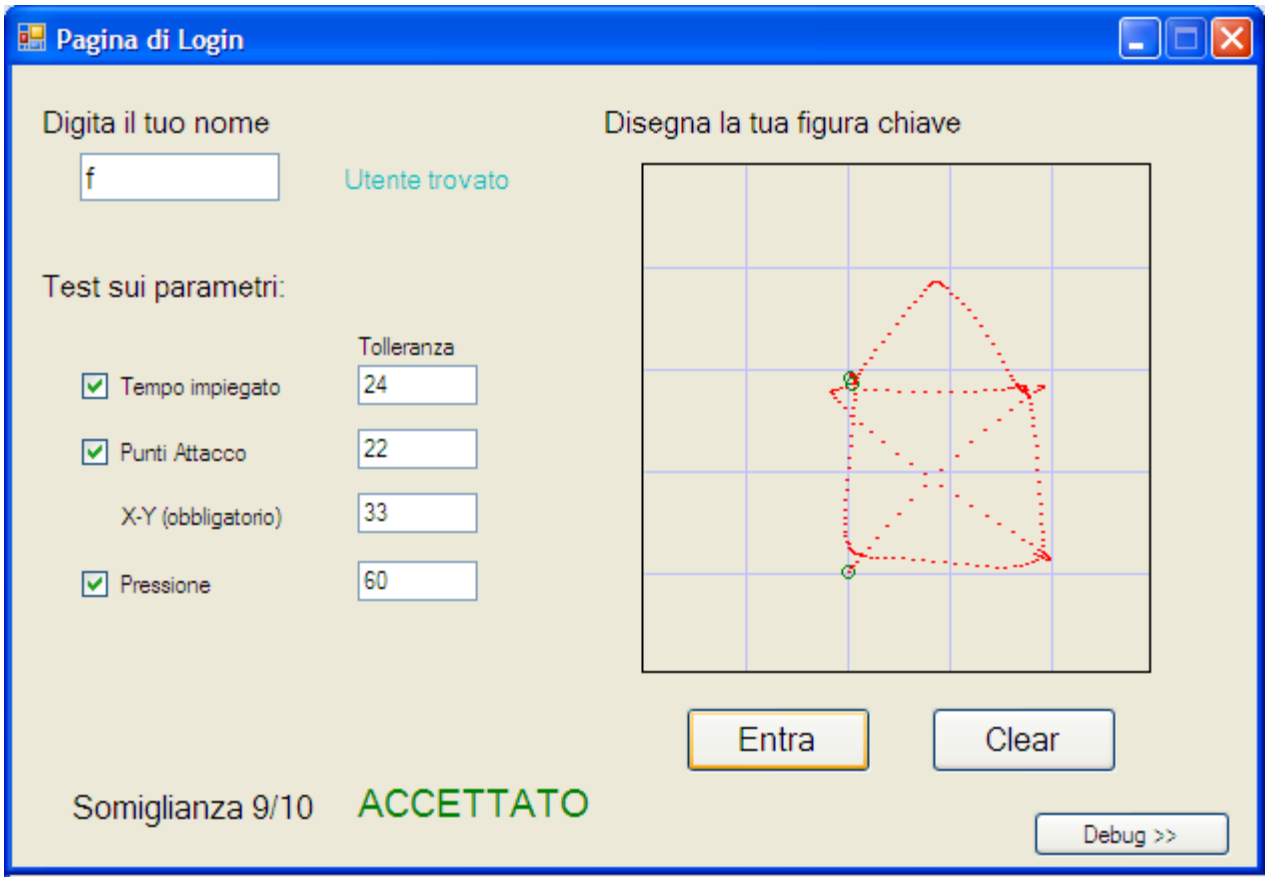

Figura 18 - Screenshot di un Login accettato

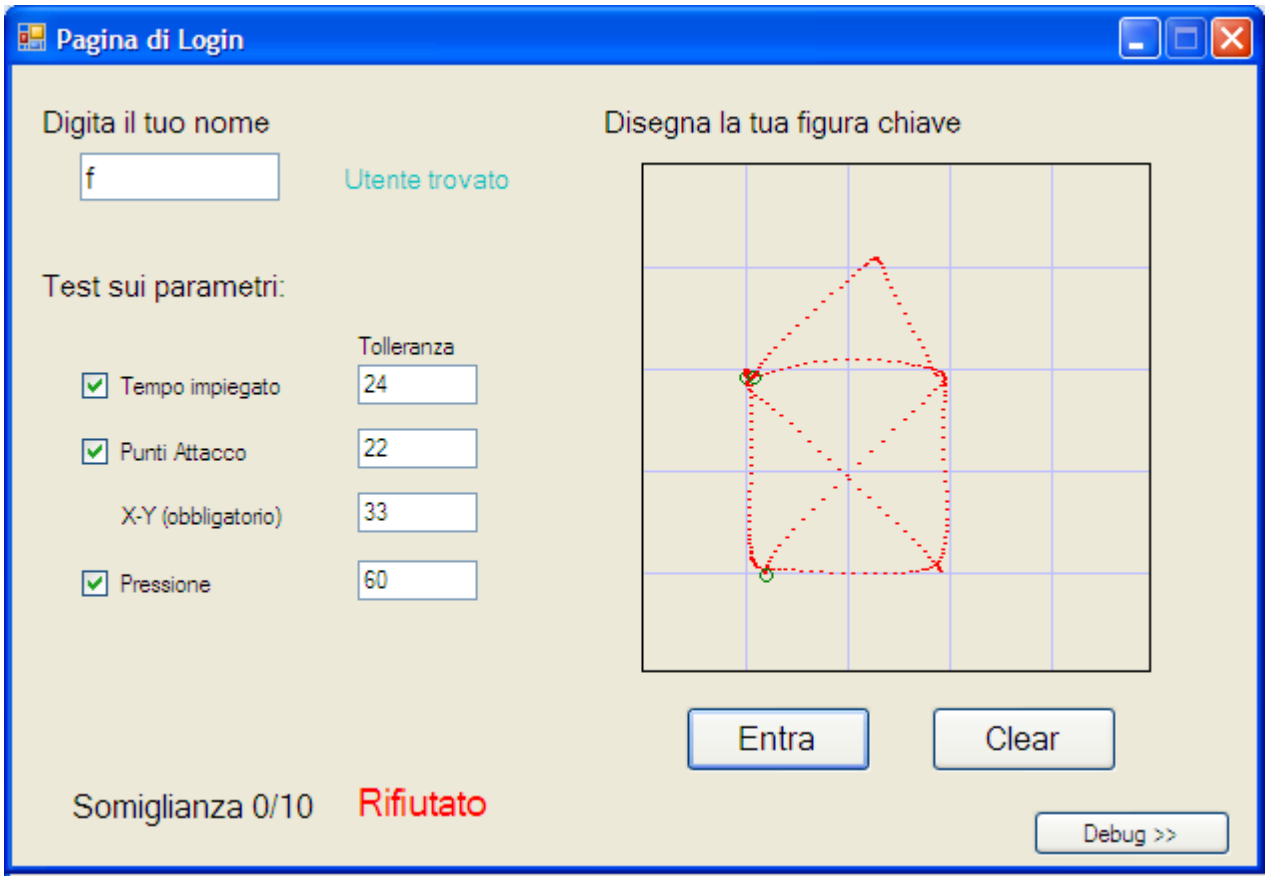

Figura 19 - Screenshot di Login rifiutato

3.4. Parametri e tolleranze : automatiche ma personalizzabili

In questo paragrafo saranno analizzati in dettaglio l'utilità e l'impostazione delle tolleranze sui confronti delle feature, che sono in sostanza i parametri regolabili da parte di un eventuale amministratore del sistema progettato.

Come già visto brevemente nei paragrafi relativi agli algoritmi di Enrollment e di Login, i valori che vengono presi come tolleranza per stabilire la somiglianza tra due esecuzioni degli stessi parametri, escluso il caso delle coordinate X e Y, sono calcolati automaticamente ed in modo personalizzato per ogni utente.

Questo, oltre a comportare l'intuitivo vantaggio di non dover effettuare tali operazioni manualmente, è un grande vantaggio in termini di flessibilità, scalabilità, sicurezza e non ultimo la facilità d'uso e di manutenzione. Scopriamo meglio il perché di questo con un esempio: un utente che è molto preciso nella posizione dei disegni ma li ripete ogni volta con tempi abbastanza diversi, ha bisogno di parametri differenti da quelli che necessita un utente che impiega sempre lo stesso tempo, ma ha il vizio di spostarsi ad ogni ripetizione rispetto alla posizione iniziale. Se ci fossero dei parametri fissi, il sistema non potrebbe adattarsi alle abitudini di ogni persona;

l'amministratore di sistema dovrebbe cambiare le tolleranze ogni volta che qualche utente deve effettuare il Login, o peggio, dovrebbe occuparsi lui stesso, tramite diversi test , di scoprire parametri adatti a ciascun utente. In alternativa, lasciare delle tolleranze fisse ma molto larghe aumenterebbe il rischio di ingresso da parte di intrusi, poiché il sistema sarebbe meno "severo".

Ecco quindi motivata la flessibilità (il sistema si adatta meglio a qualsiasi utente), la scalabilità (posso avere moltissimi utenti senza dovermi preoccupare della scelta delle tolleranze, proprio perché esse sono generate automaticamente per ogni utente), la sicurezza per quanto detto poc'anzi, e la facilità d'uso e di manutenzione visto che l'amministratore del sistema non deve impazzire sotto questo punto di vista.

Vediamo ora quali sono i parametri che si possono variare e in quali contesti.

Le feature selezionabili o deselezionabili per i confronti sono così raggruppate:

- Tempo impiegato;
- Analisi di numero e posizione dei punti di attacco;
- Analisi delle coordinate X e Y percorse (queste feature non sono disattivabili, perché renderebbero troppo scarso il sistema: infatti senza di esse sarebbe teoricamente possibile disegnare due figure completamente diverse ma caratterizzate dagli stessi punti di attacco, e se disegnate con lo stesso tempo e la stessa pressione potrebbero venir identificate come uguali);
- Analisi della pressione impiegata durante il disegno.

Le feature elencate, eccetto appunto coordinate X e Y, possono essere "disattivate". Se vengono disattivate in fase di Enrollment la relativa procedura fa a meno di catturarne i dati e di salvarli sul file del database utenti, rendendola così non attivabile al momento del Login, nella cui finestra infatti saranno oscurate le spunte corrispondenti.

Ipotizzando invece di aver registrato tutte le feature in fase di Enrollment, rimane la scelta di poterne disattivare alcune al momento del Login. E' sufficiente deselezionare le relative spunte nella finestra di Login perché l'algoritmo eviti di prendere decisioni in merito in quella sessione.

L'ultima possibilità è quella di variare le tolleranze manualmente: una volta specificato l'utente che vuole effettuare il Login infatti, vengono caricate le tolleranze personalizzate nelle apposite caselle di testo; queste possono essere variate a piacimento, se si ritiene che per uno specifico utente non siano adatte. Esempi di condizioni che possono richiedere questo intervento potrebbero essere in seguito ad Enrollment "troppo buoni", in cui cioè sono state ripetute le 10 figure in modo fatalmente troppo uguale, fornendo tolleranze molto piccole e probabilmente difficilmente rispettabili dopo qualche giorno di disuso, oppure perché un momentaneo dolore o infortunio alla mano di un certo utente potrebbe impedirgli di rispettare alcune condizioni (come ad esempio la pressione). Un'altra evenienza in cui può essere utile la variazione delle tolleranze è la volontà di variare un certo aspetto del sistema a favore di un altro: per esempio si può voler

rendere il sistema meno “severo” a favore di una diminuzione delle perdite di tempo dovute ai casi sporadici in cui l’utente può trovarsi costretto a dover far due volte il disegno per accedere al sistema perché rifiutato al primo tentativo; può essere questo il caso di ambienti in cui esistono anche altri meccanismi di verifica dell’identità, precedenti o successivi a questo.

La variazione manuale delle tolleranze non va però intesa come una pratica abituale e da adottare sempre; qualora ci siano le motivazioni per farlo, infatti, è consigliabile provvedere ad un aggiornamento dell’Enrollment. Ad esempio quando la calligrafia/disegno da parte di una persona al passare degli anni, quando è cambiato dello strumento di input (come il passaggio da una tavoletta grafica ad un tablet), o quando si verificano situazioni di Enrollment anomale o seguite da un riepilogo indicante una scarsa qualità.

4. Parte quarta: Risultati, test e conclusioni

4.1. Vantaggi teorici del sistema progettato

Questo paragrafo intende delineare quali sono i vantaggi teorici del sistema progettato in questa tesi, i quali, anche se non ancora comprovati da un intenso utilizzo del sistema (il quale richiede parecchio tempo e grande disponibilità di persone/ambienti disponibili al test), rimangono argomenti di grande rilievo; li vediamo nel seguito, suddivisi per categorie.

4.1.1. Universalità : le piattaforme adattabili

Ci eravamo posti l’obiettivo di rendere questo sistema il più possibile universale,cioè fruibile dal maggior numero di piattaforme possibili. Viste e considerate quali sono le feature richieste ed analizzate dal sistema, si può ragionevolmente affermare che questo sistema potrebbe essere utilizzato:
- In ambienti dove siano presenti Tablet PC, usandolo esattamente come nel progetto corrente, per accesso a sistemi interni aziendali (si pensi ad una azienda in cui ogni dipendente ha accesso al suo account in questo modo) o a sistemi remoti, dove un utente qualunque accede in tal modo ad un qualche sito internet ;

- In ambienti dove siano presenti dei normali PC con annessa una semplice tavoletta grafica; non essendo richiesta una precisione di digitalizzazione elevatissima, sono sufficienti delle comuni tavolette grafiche, reperibili ormai a prezzi modici. Anche in questo caso sono applicabili le due prospettive di ambiente descritte al punto precedente;

- Incorporato in dispositivi mobili touch: senza far lunghe descrizioni si pensi ad esempio ad un utente che estrae dalla tasca il suo iPhone o iPad, disegna direttamente con le dita la sua figura chiave per sbloccare lo schermo, o in sostituzione al classico PIN. Ma oltre che come misura di

sicurezza contro l'uso non autorizzato può ancora essere il caso di un sito internet a cui voglio accedere dal dispositivo mobile e infine, idea forse meno diffusa ma sicuramente possibile, un uso per l'accesso ad una qualche App personalizzata e installata su dei palmari a uso aziendale.

- Infine, se si accetta di rinunciare alla feature della pressione (di stilo, penna o dito che sia), si può utilizzare il sistema anche in un qualsiasi computer dotato di un semplice mouse; in questo caso sarà necessario che gli utenti dedichino un po' più di tempo nell'impratichirsi a far disegni possibilmente simili con il mouse, ma ciò non toglie che l'utilizzo del sistema sia ancora possibile.

4.1.2. Sicurezza

Il sistema proposto ha, in linea teorica, un vantaggio non banale in termini di sicurezza. Per chiarire come questo è possibile, immaginiamo un sistema di accesso protetto da una semplice password alfanumerica: fino a quando l'utente abilitato (o un eventuale gruppo di utenti) è l'unico a conoscere la password corretta, il sistema si può considerare al sicuro (se si ipotizza la totale assenza di bug, l'assenza di tentativi di brute forcing della password, etc.). Nel momento in cui un utente malintenzionato che riesce a raggiungere il sistema viene a conoscenza della password, tale sistema non è più al sicuro, in quanto il malintenzionato potrà entrare semplicemente digitandola.

Con il disegno di una figura chiave invece, le cose sono molto diverse. Pensiamo ancora allo stesso sistema di accesso, ma stavolta invece di una password viene chiesto di disegnare la figura chiave corretta. Ammettiamo ancora che un aggressore sia venuto a sapere qual è la figura da disegnare per entrare (ad esempio una casetta stilizzata). Stavolta però la conoscenza di questa informazione non gli è assolutamente sufficiente per accedere al sistema in quanto egli, se tenterà di disegnare la casetta, potrà essere rifiutato per numerosi motivi:

- L'aggressore difficilmente saprà in quale ordine vanno disegnati i vari tratti della figura se non ha potuto osservare attentamente "all'opera" un disegnatore autorizzato.

- Allo stesso modo è estremamente improbabile che egli sia in grado di indovinare la corretta forma da dare al disegno, la posizione precisa in cui deve essere fatto, e la grandezza che deve avere se, ancora una volta, non ha osservato il disegnatore.

- Ancora, la probabilità che egli indovini la velocità con cui deve tracciare la figura, ammesso che sia riuscito a superare quanto precedentemente descritto, è piuttosto bassa.

- Infine, anche nel caso peggiore in cui l'aggressore è riuscito ad osservare un disegnatore autorizzato ed è sufficientemente abile da imitarlo in tutti i comportamenti sopracitati, c'è ancora un ultimo ostacolo da superare, dove perfino l'osservazione nulla può: si tratta della pressione che deve essere esercitata nel fare il disegno; le diverse persone a seconda delle abitudini possono applicare una diversa pressione, eventualmente anche in singoli tratti, che altre persone non applicano.

La Figura 20 dovrebbe schematizzare in modo più immediato il minor rischio offerto da questo sistema.

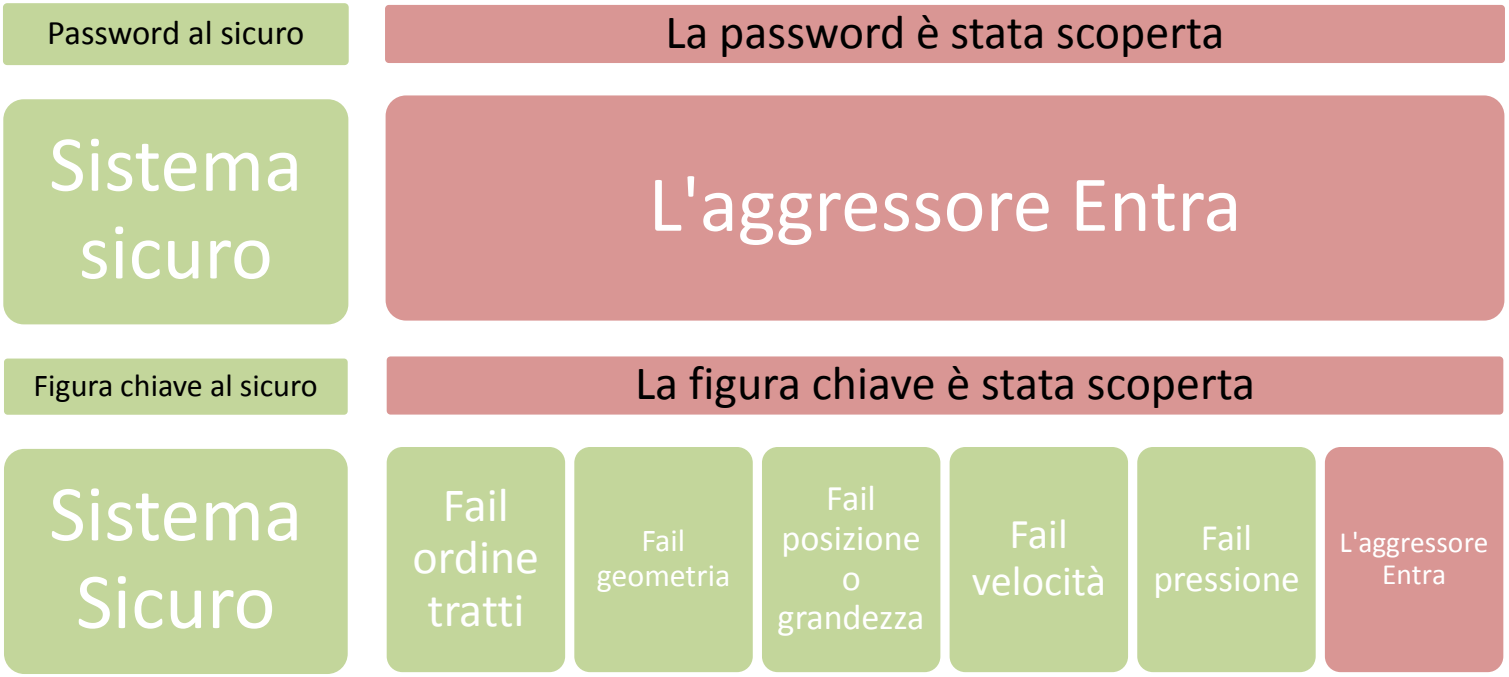

Figura 20 - Confronto tra la sicurezza offerta da un sistema protetto da password (sopra) e uno protetto tramite figura chiave (sotto)

Dall'altro lato, il rovescio della medaglia sarà l'accettazione di qualche accesso rifiutato agli stessi utenti autorizzati (rappresentato dal FAR, si vedano i relativi paragrafi).

4.1.3. Semplicità

Il sistema ha raggiunto in modo soddisfacente anche l'obiettivo della semplicità:

- In fase di Login: è sufficiente infatti specificare il proprio nome, e disegnare nell'apposito riquadro la propria figura chiave; come scrivere nome utente e password in una comune finestra di dialogo, nessuna fatica in più.
- In fase di Enrollment: anche in questo caso non ci sono grandi difficoltà, è sufficiente disegnare per 10 volte la figura chiave scelta e il sistema poi si occupa di calcolare automaticamente i parametri di tolleranza più idonei all'utente stesso, indicando pure a schermo se quanto raccolto è di qualità adeguata o se è consigliabile ripetere la procedura prestando maggiore attenzione.
- Dal lato amministratore: Il fatto che il sistema calcoli delle tolleranze specifiche per ogni utente, anziché usare dei parametri fissi, riduce l'impegno necessario anche all'eventuale amministratore del sistema, che non deve preoccuparsi di verificare tali parametri o di regolarli in funzione di ogni utente e del suo comportamento, cosa che in caso di necessità può comunque fare assicurando la massima flessibilità.
- Sull'aspetto algoritmico: esso infatti non richiede elaborazioni impegnative, ma sostanzialmente analisi, confronti e modifiche su vettori di numeri reali. Questo permette

al sistema di essere eseguito anche su dispositivi dalle basse risorse di calcolo aumentando così le possibilità di diffusione.

Un esempio di algoritmi che non godono di questo vantaggio sono talvolta gli algoritmi di analisi e riconoscimento facciale; su un normale computer possono impiegare anche più di 3 secondi per identificare un utente dato uno scatto del suo volto [23], tempo che aumenta all'aumentare dei campioni di training presenti in memoria o della risoluzione delle immagini. Questi algoritmi vedono così limitata la possibilità di un porting su dispositivi come telefoni cellulari dotati di fotocamera; un algoritmo che richiede un tempo simile su un PC da tavolo infatti, richiederebbe molto più tempo sull'hardware limitato di un cellulare, rendendo così l'esperienza di utilizzo da parte dell'utente più noiosa e meno confortevole; questo almeno sull'hardware di bassa fascia attuale, nulla toglie che in futuro le prestazioni dei dispositivi mobili aumentino in modo considerevole o che gli algoritmi migliorino.

- Ultimo breve cenno merita la semplicità intrinseca del fare un piccolo disegno stilizzato con le dita su un telefono cellulare touch, piuttosto che fare sullo stesso dispositivo una scritta o addirittura una firma, improponibile se non dotandosi di un pennino.

4.1.4. Scalabilità

Senza dilungarsi molto a riguardo, il sistema gode anche di ottima scalabilità: far funzionare il sistema con 1 solo utente è equivalente a farlo funzionare con migliaia di utenti; questo avviene, senza dare meriti immotivati al progetto, semplicemente perché nella fase di Login la decisione sull'autenticità del disegno effettuato è presa analizzando solo i dati dell'utente richiesto, senza fare confronti con i dati degli altri utenti, come fanno invece altri algoritmi oppure sistemi che prendono la decisione appoggiandosi ad un classificatore (come una Rete Neurale [24] o una Support Vector Machine [19]); il motivo di questo è spiegato nel paragrafo 2.6.2., e come già visto la scelta di non usare il classificatore permette di usare un numero pressoché infinito di simboli, permettendo quindi che gli utenti aumentino senza che sia compromessa la sicurezza e conservando la stessa breve procedura di Enrollment (dove ogni utente disegna unicamente la figura da lui scelta) qualsiasi sia il numero di utenti.

4.1.5. Strategia

A livello di "strategia" impiegata dal sistema, si integrano le tecniche dello studio della scrittura individuale con il vantaggio di non sapere quale simbolo la persona ha utilizzato come figura chiave; inoltre le figure chiavi stesse, oltre ad essere teoricamente infinite (vista la possibilità di scelta di una figura personalizzata), sono anche disegnabili in parecchi modi diversi (si veda a titolo di esempio la citazione dei 384 modi di disegnare la lettera E maiuscola nel paragrafo 2.1) sulla

differenziazione dei quali il sistema è molto attento grazie allo studio dei punti di attacco della penna.

Relativamente alle fasi di training e di utilizzo, non va dimenticato il fatto che con questo sistema non c'è bisogno di mostrare dati sensibili (figure chiave o addirittura firme) ad utenti terzi come avviene per le fasi di training in progetti come [5]. Il non dover mostrare dati sensibili a persone sconosciute favorisce l'espandibilità di utilizzo in aree eterogenee come ad esempio Internet.

4.2. Test del sistema : modalità ed esito

Questo paragrafo riporterà le modalità con cui sono stati tenuti i test eseguiti su un piccolo gruppo di persone, allo scopo di ricavare FAR e FRR del sistema progettato.

Riguardo al gruppo di persone, va premesso che:

- Sono state coinvolte 32 persone di ambo i sessi, 20 maschi, 12 femmine;
- Nessuna delle persone coinvolte aveva una precedente esperienza con tablet PC o dispositivi touch tramite penna: era quindi la prima volta che si trovavano ad adoperare uno strumento del genere, in alcuni casi con qualche difficoltà;

I test sono stati svolti con le seguenti modalità:

- A ogni utente ho spiegato per prima cosa il funzionamento e la natura del progetto, dopodiché gli ho fatto vedere una intera procedura di Enrollment e qualche tentativo di Login fatti da me;
- Ho spiegato in modo sintetico i vincoli da rispettare per l'utilizzo corretto del sistema;
- Ho chiesto all'utente di effettuare una procedura di Enrollment per sé stesso, dopodiché passa alla finestra di Login e tenta l'accesso per 20 volte. Il numero di successi e di insuccessi su questi 20 tentativi viene registrato per poter poi calcolare il FRR.
- Successivamente l'utente prova ad entrare per 20 volte nel mio account: per far ciò gli mostro in modo molto accurato come deve disegnare la mia figura chiave, in quale posizione e con che ordine di tratti per facilitarlo il più possibile nell'accesso; se noto che sbaglia a disegnare la figura chiave sempre per lo stesso motivo, gli mostro nuovamente come disegnarla bene oppure gli spiego su quale aspetto dovrebbe correggere per disegnarla meglio. Il numero di successi e di insuccessi su questi 20 tentativi viene registrato per poter poi calcolare il FAR.
- Se al termine dei 20 tentativi l'utente non è riuscito nemmeno una volta ad entrare nel mio account, gli semplifico il compito disattivando in fase di Login il controllo su una delle feature, quella su cui sbaglia più spesso, e rilevo quanti ulteriori tentativi deve fare prima di riuscire ad entrare, fermando il test appena ci riesce, oppure dopo il decimo tentativo.

Nella seguente tabella sono riportati i successi/insuccessi totali ed i relativi valori di FAR e FRR per gli utenti che hanno testato il sistema:

		Accesso al proprio account		Accesso al mio account dopo aver ricevuto precise spiegazioni su come disegnare la figura		
Utente N°	**Sesso utente**	**# Rejection**	**FRR**	**# Acceptance**	**FAR**	**Togliendo una difficoltà**
1	M	3 su 20	15,00%	0 su 20	0,00%	Ancora nessun accesso
2	M	1 su 20	5,00%	0 su 20	0,00%	Ancora nessun accesso
3	F	3 su 20	15,00%	0 su 20	0,00%	al sesto tentativo
4	M	4 su 20	20,00%	3 su 20	15,00%	
5	M	4 su 20	20,00%	0 su 20	0,00%	al primo tentativo
6	M	1 su 20	5,00%	0 su 20	0,00%	al sesto tentativo
7	M	6 su 20	30,00%	0 su 20	0,00%	Ancora nessun accesso
8	M	1 su 20	5,00%	0 su 20	0,00%	al terzo tentativo
9	M	2 su 20	10,00%	0 su 20	0,00%	al terzo tentativo
10	M	2 su 20	10,00%	4 su 20	20,00%	
11	M	2 su 20	10,00%	0 su 20	0,00%	Ancora nessun accesso
12	F	0 su 20	0,00%	3 su 20	15,00%	
13	F	4 su 20	20,00%	0 su 20	0,00%	al terzo tentativo
14	M	2 su 20	10,00%	0 su 20	0,00%	Ancora nessun accesso
15	M	5 su 20	25,00%	0 su 20	0,00%	Ancora nessun accesso
16	M	2 su 20	10,00%	6 su 20	30,00%	
17	F	5 su 20	25,00%	3 su 20	15,00%	
18	M	3 su 20	15,00%	0 su 20	0,00%	al primo tentativo
19	M	1 su 20	5,00%	1 su 20	5,00%	
20	F	0 su 20	0,00%	0 su 20	0,00%	al primo tentativo
21	M	4 su 20	20,00%	1 su 20	5,00%	
22	F	2 su 20	10,00%	2 su 20	10,00%	
23	F	4 su 20	20,00%	0 su 20	0,00%	al terzo tentativo
24	M	3 su 20	15,00%	0 su 20	0,00%	al primo tentativo
25	M	3 su 20	15,00%	0 su 20	0,00%	al primo tentativo
26	M	2 su 20	10,00%	1 su 20	5,00%	
27	F	2 su 20	10,00%	1 su 20	5,00%	
28	F	3 su 20	15,00%	2 su 20	10,00%	
29	F	4 su 20	20,00%	1 su 20	5,00%	
30	M	1 su 20	5,00%	3 su 20	15,00%	
31	F	3 su 20	15,00%	1 su 20	5,00%	
32	F	4 su 20	20,00%	1 su 20	5,00%	
		media	**13,44%**		**5,16%**	

La prima colonna indica il numero progressivo dell'utente che ha testato il sistema, la seconda colonna indica quante volte l'utente non è riuscito ad accedere al proprio account e nella terza colonna si ha il relativo FAR; la quarta colonna indica quante volte l'utente è riuscito ad entrare nel mio account dopo aver ricevuto precise istruzioni su come disegnare la mia figura chiave, nella

quinta colonna il relativo valore di FAR; nella sesta ed ultima colonna, solo per coloro che non sono riusciti nemmeno una volta ad entrare nel mio account, si ha il numero di tentativi che sono stati necessari per entrare dopo che ho disattivato il controllo sulla feature che creava più problemi a tale utente.

Per quanto riguarda il valore di FAR medio, che si attesta poco sopra il 5% (media maschile 5,83% e media femminile 4,75%), si nota che il risultato è molto apprezzabile: non solo per il valore ottenuto, ma soprattutto perché tale valore si ottiene perfino dopo aver spiegato nei minimi dettagli all'utente sotto test come dovrebbe disegnare la mia figura chiave (una operazione analoga eseguita in un sistema con password alfanumerica, porterebbe ad un FAR del 100%, perché ovviamente all'utente è sufficiente digitare tale password) , mentre in assenza di informazioni o comunque con la sola specificazione di "quale" figura chiave bisogna disegnare, il FAR è bloccato allo 0%, per la infinità di combinazioni con cui la figura può essere disegnata, posizionata,ruotata, dimensionata, all'interno del rettangolo di disegno, per la molteplicità di modi in cui disegnare i vari tratti che compongono una stessa figura, e infine per la pressoché infinità di figure disegnabili.

Per quanto riguarda il FRR invece, un valore pari a circa il 13% (media maschile 14,17% e media femminile 13,00%) indica che più di una volta su 10 l'utente medio non riesce ad entrare nel suo account, costringendolo a ripetere il disegno. Bisogna però tenere bene a mente il fatto che tutti gli utenti presi in esame non avevano mai utilizzato il dispositivo penna/tablet PC e quindi non avevano alcuna dimestichezza con esso,per cui è più che ragionevole pensare che, nel momento in cui gli utenti avranno sviluppato un po' di pratica con l'hardware, la percentuale di FRR diminuirà in modo significativo.
I più frequenti problemi che hanno causato dei "False Rejection" infatti sono stati:

- Casi di utenti che premendo troppo poco sullo schermo con la penna hanno causato dei falsi stacchi di penna, provocando un punto di attacco in più che in realtà non esiste;
- Casi di utenti che dopo aver preso un po' la mano disegnavano in modo sempre più veloce, qualche volta uscendo dai limiti di tolleranza;
- Casi di utenti che nel disegno della propria figura chiave in taluni momenti premevano molto di più rispetto alle altre volte, a causa di un momentaneo cambio di posizione del corpo, del polso o del tablet;
- Casi di utenti che stranamente avevano la tendenza a spostarsi sempre più a destra ad ogni ripetizione della figura;
- Casi di utenti che iniziavano a venire rifiutati negli ultimi dei 20 tentativi, probabilmente per una "stanchezza" dovuta al ripetere 20 volte la stessa operazione consecutivamente.
- Casi di cattiva impugnatura della penna, premendo inavvertitamente i due pulsanti ausiliari (con funzione di tasto destro e gomma) provocando interruzioni nel disegno.

La maggior parte di queste evenienze, come si sarà già intuito, può essere eliminata permettendo all'utente di impratichirsi con il tablet PC (o altro dispositivo che sia) prima di iniziare con le

procedure di Enrollment e Login e, in seguito, rilevando gli esiti di 20 tentativi di disegno ripetuti in sessioni differenti anziché tutte in una volta. Cose che in questa sessione di test non sono state possibili per ovvi motivi di tempo e di unicità del tablet PC.

4.3. Conclusioni

I risultati ottenuti dai test del sistema, citati nel precedente paragrafo, sono confortanti, in quanto pur non essendo "ottimi" dal punto di vista del FRR, lo potrebbero diventare ritestando il sistema dopo che il campione di utenti abbia maturato una certa dimestichezza con l'uso del tablet PC. Per essere stato testato su un campione di utenti presi totalmente alla sprovvista, il risultato è soddisfacente.

Le analisi e i risultati ottenuti in questa tesi mostrano quindi che l'idea seppur semplice (visto che analizza comportamenti relativamente semplici dell'individuo mentre disegna), non è banale ma ha una chance di poter essere ulteriormente studiata ed approfondita, come pure trasportata su altre piattaforme (porting).

Riguardo gli obiettivi posti all'inizio dello sviluppo, si può affermare che essi sono stati raggiunti; Siccome tutti questi quattro aspetti sono già stati citati nel paragrafo 4.1., ci si limiterà ad elencarli, rimandando al paragrafo dedicato per gli ulteriori approfondimenti:

- Universalità: intesa come la possibilità di adattare il sistema ai più svariati dispositivi (come tablet, smartphone touch, PC con tavoletta grafica) seppur non sia ovviamente ancora implementata è, come già discusso, fattibile.
- Sicurezza: come già evidenziato, il più notevole dei punti forti riscontrati è che un utente, anche se viene a conoscenza della figura chiave di un utente, e anche se si fosse perfino fatto spiegare nei dettagli come disegnarla, ha ancora scarse probabilità di accedere al sistema, il quale lo rifiuterà proprio sulla base dell'analisi comportamentale dell'individuo. Il rovescio della medaglia è che in qualche sporadico caso può succedere che un utente,sebbene autorizzato, sia rifiutato dal sistema costringendolo a ridisegnare una seconda volta la figura chiave.
- Semplicità: In fase di Login è sufficiente scrivere il proprio nome e disegnare la propria figura chiave. Analogamente a scrivere nome utente e password al momento dell'accesso ad un qualunque sito web, la procedura è quindi la più semplice possibile.
- Scalabilità: Il sistema non si appesantisce al crescere del numero di utenti registrati in esso, ma anzi, conserva le medesime prestazioni.

Tutto queste considerazioni sono basate sulle osservazioni, sui test effettuati durante la creazione del progetto e sui test effettuati sul campione di utenti. Chiaramente essendo il tutto appena nato, tali considerazioni dovrebbero essere rivedute in seguito ad un uso più intensivo del sistema.

Quel che è importante sottolineare infine, è che un sistema di riconoscimento biometrico monomodale (cioè che si basa sull'osservazione di un solo tipo di caratteristica) non è sufficientemente sicuro da poter essere adoperato da solo come sistema a difesa della sicurezza, ma è bene che le verifiche di identità siano condotte affiancando due o più analisi biometriche differenti (si parla così di una analisi multimodale). Un recente esempio che dimostra l'inadeguatezza dei sistemi monomodali è la vulnerabilità scoperta da un gruppo di ricercatori riguardo il sistema di riconoscimento facciale integrato sui più recenti PC portatili dotati di webcam (maggiori dettagli in [25]). Questi ricercatori sono infatti riusciti a violare il sistema sui tutte e tre le marche di PC portatili che lo integravano (Asus, Toshiba, Lenovo) semplicemente utilizzando una foto della persona autorizzata all'accesso. Leggere risultati simili quindi, su una branca della biometria come il riconoscimento facciale che negli ultimi tempi sta riscuotendo grandi attenzioni, suggerisce che sia sempre meglio affidarsi a sistemi multimodali per garantire un adeguato livello di sicurezza.

4.4. Possibili sviluppi e sbocchi futuri

4.4.1. Confronto basato su media-varianza di variabile aleatoria

I dati ottenuti dall'osservazione delle varie feature potrebbero essere modellati tramite una qualche variabile aleatoria (per molte feature si potrebbe ipotizzare una gaussiana), caratterizzata da una propria media ed una propria varianza. Successivamente le elaborazioni necessarie per l'accettazione o meno dell'utente in fase di Login potrebbero essere fatte, anziché confrontando tutti i dati, valutando se taluni parametri rientrano in intervalli definiti tramite le predette media e varianza. Tale scelta comporta una modifica abbastanza radicale degli algoritmi attuali; in questo paragrafo spiegheremo come sia possibile raggiungere questo obiettivo, in modo da facilitare il lavoro in una futura implementazione.

- Per quanto riguarda l'algoritmo di Enrollment è necessario modificarlo nel modo seguente: Dopo aver ricevuto dall'utente i dati dei 10 campioni disegnati, bisogna calcolare media e varianza per ciascuna delle feature considerate.
    - Per le feature la cui analisi restituisce uno o più valori interi o reali (per sapere quali sono si rimanda al paragrafo 2.4.), la modifica consiste in: calcolare la media M (eventualmente pesata con un vettore di pesi P(i) con 1 <= i <= 10 , dove il generico P(i) rappresenta il peso attribuito all'i-esimo campione) e varianza V dei 10 valori della feature considerata. Dopodichè ci si limita a salvare su file solamente queste 2 grandezze M e V anziché tutti e 10 i valori campionati.
    - Per le feature la cui analisi restituisce un vettore V(x) con 0 < x < 350 (si rimanda nuovamente al paragrafo 2.4.), la modifica consiste nel calcolare media e varianza di ogni punto x nei dieci vettori a disposizione, dando così origine a due nuovi

vettori M(x) e Var(x) anch'essi con 0 < x < 350. Dopodichè si salvano su file solamente i vettori M(x) e Var(x).

- Per quanto riguarda l'algoritmo di Login invece le modifiche sono più interessanti. Dopo aver campionato e normalizzato la figura chiave di Login, si procede ancora una volta al confronto delle feature da essa ottenute con i dati forniti dalla fase di Enrollment per tale utente.
    - Per le feature la cui analisi restituisce uno o più valori interi o reali, dato il valore L ottenuto dal campione di Login, e i valori M e V corrispondenti a media e varianza a disposizione, si controlla se L è contenuto nell'intervallo (dipendente dalla variabile aleatoria scelta) definito dai parametri M e σV dove sigma diventa un parametro variabile dal progettista a seconda delle necessità. A questo punto per decidere se il valore L è accettabile oppure no, il progettista avrà varie possibilità, come ad esempio accettare L se rientra nell'intervallo [M-σV , M+σV] con σ fissato e rifiutarlo in caso contrario, oppure attribuire ad L un punteggio di bontà P, che assuma il massimo valore quando L è compreso nel predetto intervallo con σ=1 e che scenda gradualmente all'aumentare di σ; le possibilità su questo aspetto sono molteplici, quindi sta alla fantasia del progettista che l'implementerà prendere la decisione.
    - Per le feature la cui analisi restituisce un vettore V(x) con 0 < x < 350, si dovrà verificare che per ogni punto x del vettore, dove 0 < x <350, V(x) sia contenuto nell'intervallo [M(x)-σV(x) , M(x)+σV(x)] . Anche in questo caso una (ma non l'unica) possibile modalità per effettuare la valutazione è quella di dare un punteggio P_x ad ogni punto, massimo quando i punti sono contenuti nell'intervallo con σ=1 e decrescente all'aumentare di σ. Si procederà poi alla somma di tutti i punteggi P_x con 1<= x <=350 per ottenere un valore S da giudicare a discrezione del progettista.

Glossario dei termini

- Enrollment : fase in cui il sistema acquisisce i dati di cui ha bisogno per riconoscere/verificare l'identità dell'utente: è sinonimo di procedura di registrazione.
- Esecuzione : si veda il sinonimo "Ripetizione".
- Feature : nel testo a volte indicata anche come "Caratteristica", è uno degli aspetti di interesse rilevati dal sistema, cioè una delle grandezze fisiche osservate mentre l'utente compone il disegno.
- Figura chiave : Un disegnino, stilizzato o meno, una sigla oppure uno schizzo, disegnato in modo naturale e sbrigativo, analogamente a una firma, senza l'obiettivo di ottenere precisione o bellezza ma con il solo obiettivo di fungere da "password" per l'accesso al sistema progettato e descritto in questa tesi.
- Punto di attacco : è il punto in cui la penna (di una tavoletta grafica o di un Tablet PC) si appoggia sullo schermo all'inizio del disegno di un tratto. Si può chiamare in modo equivalente "punto di Landing". A titolo di esempio, quando si disegna una X con la penna, si generano due punti di attacco, uno quando si inizia a disegnare il primo tratto, il secondo quando, dopo aver alzato la penna, la si riappoggia sulla superficie per disegnare il rimanente tratto.
- Ripetizione : In questo contesto, si indica con "ripetizioni" o "ripetizioni di un disegno" o "ripetizioni di una figura chiave", l'atto di disegnare più volte una stessa figura chiave, consecutivamente o in momenti distinti.

Elenco delle figure

Bibliografia

[1] Lambert Schomaker, "Advances in Writer identification and verification".

[2] L. Hong, and S. Pankanti A. Jain, "Biometric identification.," 2000.

[3] Wikipedia. Biometrics and its performance metrics. [Online]. http://en.wikipedia.org/wiki/Biometric

[4] L. Schomaker and M. Bulacu, "Automatic writer identification using connected-component contours and edge-based features of upper-case western script.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004.

[5] Julian Fierrez Javier Ortega-Garcia,.

[6] E. Okamoto, M. MAMBO K. Hayashi, "Proposal of user identification scheme using mouse," 1998.

[7] Anoop Namboodiri and Sachin Gupta, "Text Independent Writer Identification from Online Handwriting," *(International Institute of Information Technology)*.

[8] BioID HumanScan. About FAR,FRR and EER. [Online]. http://support.bioid.com/sdk/docs/About_EER.htm

[9] Definizione di FAR e FRR e valori tipici nel riconoscimento delle impronte digitali. [Online]. http://www.infordata.it/support/index/detail/id/8/title/Cosa_il_FAR_e_cosa_il_FRR

[10] Eiji Okamoto, Masahiro Mambo Agus Fanar Syukri, "A User Identification System Using Signature Written With Mouse," *School of Information Science, Japan*.

[11] "Il tablet PC italico". (2009, Giugno) Tecnologie utilizzate nei digitalizzatori dei touch screen. [Online]. http://www.italico.tabletpc.it/2009/06/breve-guida-alla-tecnologia-dei.html

[12] www.wacom.com Wacom official site. Driver aggiornato per digitalizzatori Wacom , v.5.05.

[Online]. ftp://ftp.wacom-europe.com/pub/WINDOWS/cons5.05-7_int.exe

[13] Microsoft Windows XP Tablet PC Edition SDK 1.7. [Online]. http://www.microsoft.com/downloads/en/details.aspx?FamilyID=b46d4b83-a821-40bc-aa85-c9ee3d6e9699

[14] Libreria MSDN sulla programmazone dei Tablet PC. [Online]. http://msdn.microsoft.com/en-us/library/ms698573(v=VS.85).aspx

[15] Libreria MSDN : API per il Tablet PC. [Online]. http://msdn.microsoft.com/en-us/library/ms696350(v=VS.85).aspx

[16] Libreria MSDN : Classe RealTimeStylus per l'analisi a basso livello dei dati forniti dalla penna del Tablet PC. [Online]. http://msdn.microsoft.com/en-us/library/ms701683(v=VS.85).aspx

[17] Libreria MSDN : Guida alla classe RealTimeStylus. [Online]. http://msdn.microsoft.com/en-us/magazine/cc300776.aspx#S3

[18] DevX.com : Guida all'uso della classe RealTimeStylus. [Online]. http://www.devx.com/codemag/Article/30352/1954

[19] Wikipedia. Support Vector Machines. [Online]. http://en.wikipedia.org/wiki/Support_vector_machine

[20] LIBSVM. Un classificatore SVM: download, guida all'uso e alla regolazione dei parametri. [Online]. http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[21] Classification types in SVM (especially about nu-SVC). [Online]. http://scikit-learn.sourceforge.net/modules/svm.html

[22] More on classification techniques and nu-SVC. [Online]. http://www.statsoft.com/textbook/support-vector-machines/

[23] Karl B. J. Axnick and Kim C., "Fast Face Recognition".

[24] Wikipedia. Rete Neurale (it). [Online]. http://it.wikipedia.org/wiki/Rete_neurale

[25] DarkReading. Researchers Hack Faces In Biometric Facial Authentication Systems. [Online]. http://www.darkreading.com/security/vulnerabilities/213901113/index.html