



UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI MATEMATICA  
CORSO DI LAUREA IN INFORMATICA

# Classificazione firme statiche utilizzando i Hidden Markov Models

*Relatore:*  
Ch.mo Prof. Tullio VARDANEGA

*Laureando:*  
Alexandru PRIGOREANU  
1004887

Anno accademico 2012/2013

---

## Sommario

La presente relazione ha come scopo la descrizione dell'attività di stage, svolta dal sottoscritto, nel periodo settembre-ottobre 2013 presso l'azienda Corvallis. Il primo capitolo descrive l'azienda ospitante. Il secondo capitolo espone le motivazioni e gli obiettivi del progetto di stage. Il terzo capitolo illustra in modo approfondito le attività effettuate per raggiungere gli obiettivi prefissati. Il quarto ed ultimo capitolo riporta una valutazione a posteriori sul lavoro svolto, sulle conoscenze acquisite e sulla distanza tra le conoscenze richieste e le conoscenze possedute.

---

## Convenzioni tipografiche

Al fine di migliorare la leggibilità e la chiarezza dei contenuti esposti sono state adottate le seguenti convenzioni tipografiche:

- Problema : pdf ci sono i link andata e ritorno ... ma stampato no!, definisci convenzioni che valgono per entrambe le situazioni
- 
-

## Indice

<b>1</b>	<b>Dominio applicativo</b>	<b>4</b>
1.1	Azienda . . . . .	4
1.2	Prodotti e clienti . . . . .	4
1.2.1	Banche . . . . .	4
1.2.2	Assicurazioni . . . . .	5
1.2.3	Industria e servizi . . . . .	5
1.2.4	Pubblica Amministrazione . . . . .	5
1.2.5	Know Your Customer (KYC) . . . . .	6
1.3	Tecnologie . . . . .	6
1.4	Processi interni . . . . .	6
<b>2</b>	<b>Progetto aziendale</b>	<b>8</b>
2.1	Introduzione alla classificazione di firme statiche . . . . .	8
2.1.1	Valutazione delle performance . . . . .	9
2.1.2	Tipi di falsificazione . . . . .	9
2.1.3	Verifica della firma statica (offline signature verification) . . . . .	10
2.1.4	Prototipo preesistente . . . . .	13
2.2	Obiettivi . . . . .	18
2.3	Aspettative . . . . .	18
2.4	Vincoli . . . . .	18
<b>3</b>	<b>Attività di stage</b>	<b>19</b>
3.1	Pianificazione . . . . .	19
3.2	Analisi . . . . .	19
3.2.1	Ricerca di migliorie del prototipo preesistente . . . . .	19
3.2.2	Scelta di un nuovo classificatore di firme statiche . . . . .	19
3.2.3	Requisiti . . . . .	19
3.3	Progettazione . . . . .	19
3.4	Implementazione . . . . .	19
<b>4</b>	<b>Valutazione retrospettiva</b>	<b>20</b>
4.1	Copertura dei requisiti . . . . .	20
4.1.1	Possibili sviluppi alle attività svolte . . . . .	20
4.2	Conoscenze acquisite . . . . .	20
4.3	Distanza tra conoscenze richieste e conoscenze possedute . . . . .	20
	<b>Glossario</b>	<b>21</b>
	<b>Riferimenti</b>	<b>22</b>

## Elenco delle tabelle

1	Risultati test prototipo preesistente . . . . .	18
---	---	----

## Elenco delle figure

1	Logo azienda Corvallis . . . . .	4
2	Tecnologie in uso . . . . .	7
3	Sviluppo Software, ciclo di vita . . . . .	8
4	Tipi di falsificazioni: . . . . .	10
5	Workflow dei sistemi di verifica della firma statica . . . . .	11
6	Features statiche e features pseudo-dinamiche . . . . .	12
7	Panoramica metodi di classificazione . . . . .	13
8	Il processo . . . . .	14
9	Training per firmatario . . . . .	16
10	Verifica firma . . . . .	17

## 1 Dominio applicativo

La prima parte di questa relazione si prefigge di presentare al lettore il contesto di lavoro dell'azienda ospitante il progetto di stage. Inizialmente descrivo brevemente l'azienda. A seguire effettuo una panoramica sui prodotti e servizi che essa offre e sui clienti che cerca di soddisfare. Infine elenco alcune delle tecnologie di appoggio allo sviluppo software e i processi interni di quest'ultimo.

### 1.1 Azienda



Figura 1: Logo azienda Corvallis

Corvallis S.p.A. (<http://www.corvallis.it/>) è una società italiana presente nel settore dell'Information Technology. Essa applica le sue competenze funzionali, tecnologiche e di processo, acquisite in quasi trenta anni di operatività, in 4 settori strategici per il mondo finance:

- risparmio gestito e crediti;
- document management;
- area compliance;
- evoluzione tecnologica.

Corvallis fornisce prodotti e servizi a clienti appartenenti all'ambito finanziario (banche e assicurazioni), industria/servizi e Pubblica Amministrazione. In particolare, Corvallis effettua attività di progettazione e sviluppo software, di fornitura di prodotti software propri e di terzi, di manutenzione e assistenza tecnica, di consulenza.

Il Gruppo Corvallis è composto da più di 600 risorse, tra dipendenti e collaboratori, distribuite in 13 sedi operative nel territorio italiano.

### 1.2 Prodotti e clienti

#### 1.2.1 Banche

Corvallis opera da molti anni con i principali istituti di credito e società prodotto. Le aree di competenza sviluppate riguardano:

- Risparmio Gestito;
- Finanza;
- Crediti;
- Compliance;

- Governance;
- Sistemi di Pagamento;
- Document Management.

### **1.2.2 Assicurazioni**

Corvallis può effettuare outsourcing nelle compagnie Assicuratrici per i processi Vita e Danni (IT e backoffice). L'offerta verso le compagnie assicuratrici riguarda le seguenti aree:

- Sistema Portafoglio Vita e Danni;
- Sistemi di agenzia;
- Sistemi sinistri;
- Compliance;
- Governance;
- Servizi di Back Office;
- Outsourcing;
- Document Management.

### **1.2.3 Industria e servizi**

I clienti di Corvallis sono alcune industrie e imprese appartenenti a differenti settori: Oil & Gas, Engineering, Construction, Aerospace & Defense, Telco, Media. Corvallis possiede competenze anche nell'ambito del Project & Portfolio Management. Inoltre svolge erogazione di servizi di supporto specialistico relativo a prodotti leader di mercato tra i quali Cognos, Qlikview, Hyperion, Business Object e IridionDQ.

### **1.2.4 Pubblica Amministrazione**

Corvallis offre soluzioni e servizi alle pubbliche amministrazioni. Alcune di queste riguardano:

- Tributi e fiscalità per gli enti locali;
- Sistemi informativi territoriali e catastali, GIS, Digital Mapping e cartografia;
- Document Management;
- Sistemi Informativi per la catalogazione, conservazione e la tutela dei Beni Culturali.

### 1.2.5 Know Your Customer (KYC)

Know Your Customer è un prodotto sviluppato dall'azienda Corvallis per soddisfare i requisiti normativi richiesti dalla direttiva Cee in materia di Antiriciclaggio. È un software utile alle banche e alle assicurazioni poiché permette di tenere sotto controllo la segnalazione di operazioni sospette previste dalla normativa. Il prodotto si compone di sette moduli integrabili e flessibili:

- Repository, modulo che archivia in un unico ambiente tutte le informazioni relative al cliente e la sua operatività, necessarie a soddisfare le diverse richieste imposte dalla normativa;
- Classifier, modulo che racchiude una generale profilatura del Cliente rispetto ad un numero non prefissato di dimensioni, indipendenti tra loro ma con la condivisione di dati e funzioni;
- Integrator, modulo di integrazione delle fonti esterne, fornisce tutte le funzioni richieste per l'import e l'aggregazione delle liste sia pubbliche che private (ad esempio Persone Politicamente Esposte);
- Expansion, modulo basato su logiche di intelligenza artificiale, soddisfa l'esigenza di ricerche di ricorrenze di dati, presenti nelle liste pubbliche, nei database interni, nei database forniti da info providers, oppure nei dati acquisiti tramite il modulo Integrator;
- Control, modulo che presidia l'analisi dell'operatività della clientela considerando i dati anagrafici, di rapporto e di movimentazione, generando degli aggregati statistici da analizzare;
- Monitor, modulo che propone un'interfaccia utente per visualizzare e analizzare gli alert generati dai moduli Classifier, Expansion e Control;
- Configurator, modulo che permette di assicurare un approccio flessibile ed efficiente che può essere facilmente modificato rispondendo ai cambiamenti repentini delle regole internazionali o interne all'organizzazione. Il modulo governa le parametrizzazioni delle altre componenti del prodotto KYC.

Il software è stato scritto in Java utilizzando il framework aziendale.

## 1.3 Tecnologie

Seguono alcune delle tecnologie impiegate nello sviluppo software.

I linguaggi di programmazione più utilizzati sono Java e Cobol.

Il sistema di versionamento in uso è il Concurrent Versions System (CVS).

Alcuni dei database utilizzati sono: DB2, Oracle, MS SQL, PostgreSQL

I framework maggiormente utilizzati sono JBoss e JWolf. JWolf è stato creato dall'azienda stessa. È un framework per lo sviluppo di applicazioni su tecnologia Java multiplatforma e multicanale e racchiude le best practices dell'architettura Service Oriented Architecture (SOA).

L'ambiente di sviluppo per Java è Eclipse.

In figura 2 possiamo vedere uno schema riassuntivo di alcune delle tecnologie in uso.



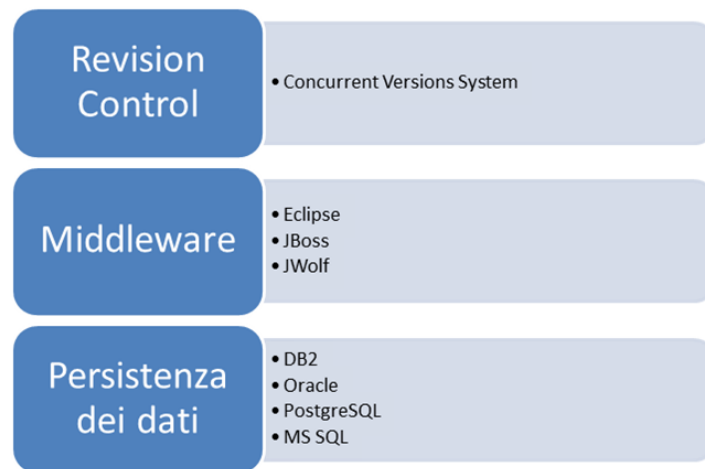


Figura 2: Tecnologie in uso

## 1.4 Processi interni

Le fasi di sviluppo di un progetto sono costituite da:

- **Coordinamento e Riunioni.** In questa fase vengono pianificate tutte le attività necessarie allo svolgimento del progetto. Gli incontri con i clienti hanno come scopo una efficiente trasmissione di informazioni;
- **Analisi dei requisiti.** L'output dell'attività di analisi è un documento in cui vengono racchiusi tutti i requisiti funzionali, qualitativi, prestazionali e dichiarativi dei quali il prodotto finale dovrà garantirne il soddisfacimento. Il documento serve da input per la fase di progettazione;
- **Progettazione.** Nella fase di progettazione si definiscono le specifiche tecniche delle funzionalità da realizzare. Il risultato di questa fase è il documento di Specifiche Tecniche di Progettazione;
- **Sviluppo.** È lo stadio esecutivo del progetto con il quale si realizzano i moduli software previsti dal disegno applicativo. In questa fase vengono effettuati test di unità;
- **Test funzionali e di sistema.** I test funzionali hanno lo scopo di verificare che i moduli realizzati durante la fase di codifica rispettino quanto fissato dai requisiti iniziali. Il test di sistema valida il prodotto nella sua interezza;
- **Collaudo con il cliente.** Si tratta di un test di sistema effettuato su un ambiente del cliente e con dati di prova forniti dallo stesso. L'output di questa attività è un verbale che racconta l'esito del collaudo;
- **Documentazione di prodotto.** Questa fase prevede la stesura dei Manuali di Prodotto relativi al software realizzato;

L'output di ogni fase viene verificato e, se conforme agli standard di qualità dell'azienda, approvato. Altrimenti dovranno essere indicate delle misure correttive per i problemi individuati.

In figura 3 vediamo un resoconto delle fasi necessarie allo sviluppo software.



Figura 3: Sviluppo Software, ciclo di vita

## 2 Progetto aziendale

Questo capitolo ha lo scopo di mostrare al lettore l'ambito in cui si colloca il progetto di stage proposto dall'azienda Corvallis. Inizialmente presento i motivi che hanno portato alla nascita del progetto. A seguire effettuo una panoramica sulla classificazione delle firme statiche. In seguito descrivo il prototipo/classificatore preesistente al progetto di stage. Infine illustro gli obiettivi, le aspettative e i vincoli del progetto aziendale a cui ho preso parte.

Visti i notevoli vantaggi in termini di incremento dell'efficienza e di riduzione dei costi che la [dematerializzazione](#) garantisce (risparmio relativo ai costi di stampa, acquisto e manutenzione delle stampanti), nell'ambito delle nuove normative alle banche sarà concesso lo scambio di immagini degli assegni bancari.

In base a questa premessa l'azienda Corvallis ha intuito che il core di un'applicazione bancaria che accetta lo scambio di immagini degli assegni bancari sarà un classificatore di firme statiche accurato, robusto e affidabile. Conseguentemente ha implementato un prototipo per la classificazione di firme statiche.

### 2.1 Introduzione alla classificazione di firme statiche

La firma è un tratto comportamentale di un individuo e costituisce una particolare classe di scrittura dove lettere o parole possono essere non distinguibili. È considerata un elemento distintivo avente caratteristiche uniche e personali. Accertare in maniera chiara ed univoca il sottoscrittore di un documento avente forza legale (in questo caso assegni bancari) è di fondamentale importanza. Infatti il destinatario deve poter identificare l'identità del mittente (autenticità) e il mittente non deve poter disconoscere un documento da lui firmato (non ripudio). Nasce quindi l'esigenza di distinguere tra firme false e firme autentiche.

Le difficoltà principali nel classificare le firme autografe sono dovute alle variazioni intrapersonali: le firme di una persona possiedono grande variabilità, dovuta allo stato emotivo dei sottoscrittori oppure alla posizione di raccolta, e di conseguenza, se confrontassimo due esemplari di firma di un firmatario questi non sarebbero identici. Invece l'agevolazione cardinale sta nel fatto che le firme di persone diverse manifestano caratteristiche elementari distinte.

A seconda del hardware front-end, un sistema di verifica della firma (signature verification system) può essere etichettato come offline o online. Nei sistemi offline la verifica della firma avviene dopo la sottoscrizione della firma. Le uniche informazioni che si possiedono sono di natura statica: l'immagine della firma autografa del sottoscrittore. Al contrario, nei sistemi online, le firme vengono acquisite tramite un dispositivo elettronico (tavola grafica) oppure con una penna speciale, capace di memorizzare una sequenza di punti che descrivono velocità, pressione, ritmo, accelerazione e movimento effettuati dal sottoscrittore, non solo l'immagine statica.

#### 2.1.1 Valutazione delle performance

La performance di un sistema di verifica della firma si calcola in base alla percentuale di errori che esso commette nel classificare le firme. Esistono due tipi di errori. Questi vengono riassunti dai seguenti indici:

- False Acceptance Rate (FAR), indice di accettazione dei falsi, ossia percentuale delle firme false classificate come genuine;

$$FAR = \frac{nr. \text{ falsi accettati}}{nr. \text{ falsi totali}}$$

- False Rejection Rate (FRR), indice di rifiuto dei genuini, ossia percentuale delle firme genuine classificate come false.

$$FRR = \frac{\text{nr. genuini rifiutati}}{\text{nr. genuini totali}}$$

Purtroppo i due indici sono inversamente proporzionali: a una diminuzione del FAR corrisponde un aumento del FRR e viceversa a una diminuzione del FRR corrisponde un aumento del FAR. È evidente quindi che si deve arrivare a un compromesso: minimizzare il FAR conservando un valore tollerabile per il FRR.

Se si eguagliano i due indici si è in presenza di un nuovo indice: Equal Error Rate (EER). L'EER permette di paragonare in modo veloce l'accuratezza dei sistemi di verifica. In generale, più l'indice EER è basso più il sistema è accurato.

Non sorprendentemente, grazie ai dati biometrici che i sistemi di verifica della firma online possiedono in più, essi sono più accurati e affidabili dei sistemi di verifica della firma statica.

### 2.1.2 Tipi di falsificazione

In letteratura sono stati individuati tre tipi di falsificazione a seconda del grado di preparazione del falsificatore sulla firma che sta cercando di riprodurre:

- falsificazioni casuali (random forgeries), prodotte senza conoscere né il nome del firmatario né la forma della sua firma;
- falsificazioni semplici (simple forgeries), prodotte conoscendo il nome del firmatario ma senza avere un esempio della sua firma;
- falsificazioni accurate (skilled forgeries), prodotte dopo un allenamento con l'obiettivo di imitare la firma originale nel miglior modo possibile.

In figura 4 vediamo un esempio di firma genuina e degli esempi di falsificazione della stessa.



Figura 4: Tipi di falsificazioni:  
(a) firma genuina; (b) falsificazione casuale;  
(c) falsificazione semplice; (d) falsificazione accurata

### 2.1.3 Verifica della firma statica (offline signature verification)

La verifica di firme statiche appartiene alla classe di problemi del pattern recognition e un tipico sistema di pattern recognition si compone dai seguenti step [1]:

1. Acquisizione dati, cattura delle immagini firma;
2. Preprocessing, per semplificare le operazioni successive minimizzando la perdita di informazioni;
3. Estrazione delle caratteristiche (feature extraction), riduzione dei dati in input misurando solo determinate caratteristiche (feature) o proprietà;
4. Classificazione (oppure fase di verifica), prendere una decisione (accettare o rifiutare) sulla base dell'adeguatezza dei valori ottenuti dalla fase di feature extraction.

In figura 5 vediamo il workflow generale del processo di verifica della firma statica.

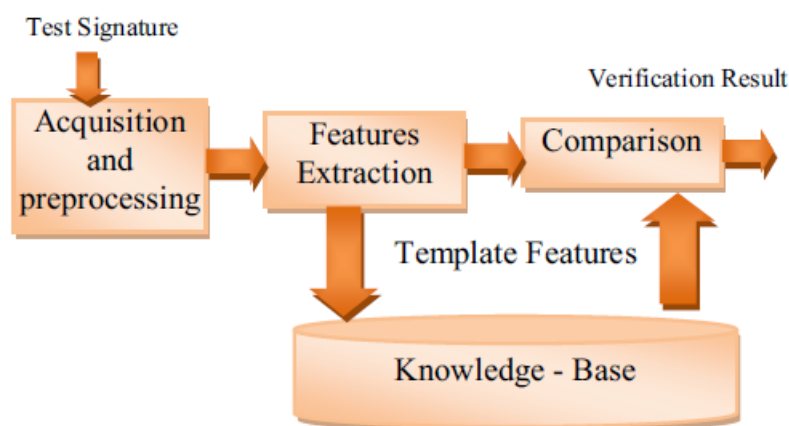


Figura 5: Workflow dei sistemi di verifica della firma statica  
Sorgente: [4]

### Acquisizione dati

Le immagini di firma vengono scannerizzate oppure vengono estratte da documenti PDF.

### Preprocessing

Il preprocessing della firma è uno step fondamentale necessario a migliorare l'accuratezza dell'estrazione delle caratteristiche e della classificazione. Lo scopo della fase di preprocessing è quello di standardizzare le firme e renderle pronte per la fase di feature extraction.

Di seguito elenco alcune delle operazioni di preprocessing più utilizzate:

- Estrazione della firma (signature extraction);
- Ritaglio (cropping), l'immagine viene ridotta al rettangolo che circoscrive la firma;
- Ridimensionamento (resizing), l'immagine della firma viene ridimensionata;
- Binarizzazione (binarization), conversione da colore a scala di grigi e infine a bianco-nero (binario);
- Assottigliamento (thinning), assottigliamento del tratto della firma a un pixel per uniformare lo spessore;

Un'errata scelta degli algoritmi di preprocessing può comportare una considerevole perdita di informazione.

### Estrazione delle features

Il successo di un sistema di verifica della firma dipende fortemente dalla fase di Estrazione delle features. Un metodo ideale di feature extraction prevede l'estrazione di quelle caratteristiche che minimizzano le variazioni intrapersonali presenti nelle firme di un sottoscrittore e massimizzano le variazioni interpersonali presenti nelle firme di due sottoscrittori distinti[2].

A seconda della facilità con la quale si possono imitare, le feature possono essere classificate in feature statiche o feature pseudo-dinamiche. La prima categoria contiene caratteristiche percettive e quindi facili da imitare, mentre la seconda categoria contiene caratteristiche poco intuitive e quindi di difficile imitazione[3]. Segue un elenco stringato delle features più utilizzate in letteratura.

Features statiche:

- Calibre, valore che individua il rapporto tra altezza e larghezza della firma;
- Proportion, si riferisce alle simmetrie presenti nell'immagine
- Spacing, valore che individua il numero di blocchi che compongono la firma, se non vi sono spaziature questo valore è uguale a 1;
- Base behavior, descrive l'angolo di inclinazione della firma in base a un'immaginaria linea orizzontale.

Features pseudo-dinamiche:

- Density of pixels, descrive la larghezza dei tratti, viene considerata pressione apparente;
- Distribution of pixels, descrive il numero di pixel neri presenti in uno spazio;
- Slant, individua informazioni riguardo all'inclinazione del tratto in considerazione;
- Form, individua informazioni riguardo alla forma del tratto in considerazione;

Un'ulteriore suddivisione delle features è dovuta alla segmentazione. In presenza di un'immagine firma segmentata le features possono essere globali (riguardano l'immagine nella sua interezza) o locali (riguardano determinate parti dell'immagine).

In figura 6 vediamo la panoramica delle features sopra elencate.

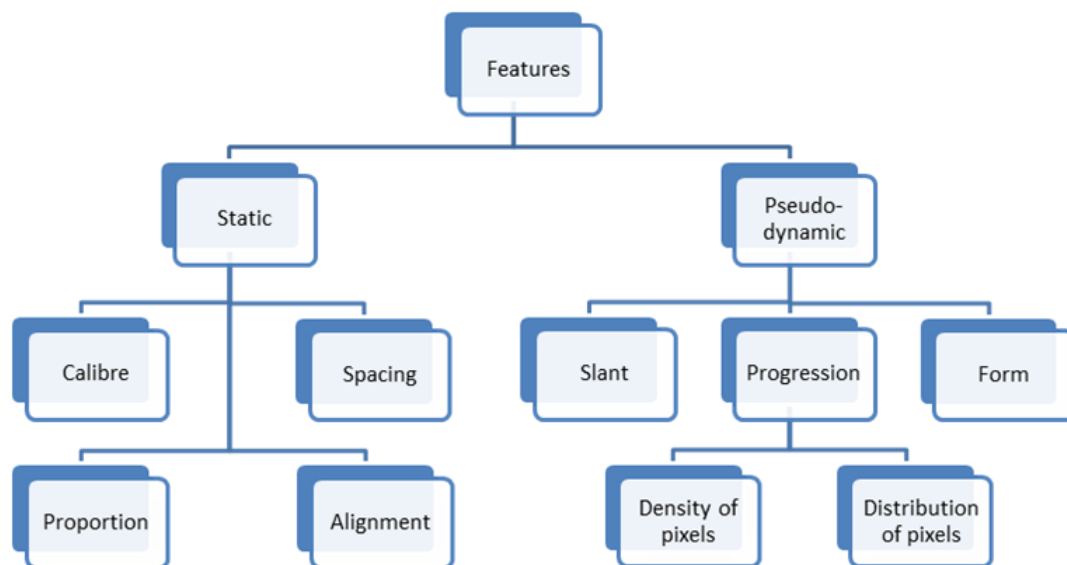


Figura 6: Features statiche e features pseudo-dinamiche

### Classificazione

L'ultimo step nel processo di verifica dei sistemi offline decide se la firma presa in considerazione è genuina o falsa.

In letteratura esistono diversi tipi di approcci al problema della classificazione di firme statiche.

In figura 7 riporto un grafico riassuntivo degli approcci più utilizzati.

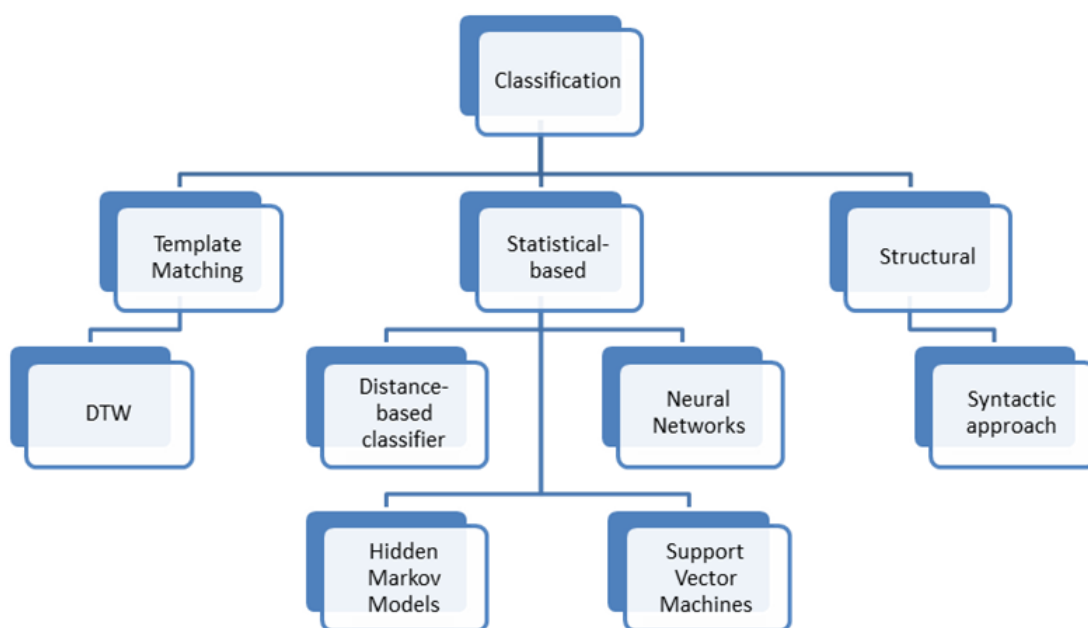


Figura 7: Panoramica metodi di classificazione

Poiché alcuni dei prossimi argomenti di questa relazione trattano i classificatori di tipo Statistical-based, né spiego la logica.

### Classificatori Statistical-based

Nei classificatori di tipo Statistical-based la conoscenza statistica viene utilizzata per usufruire di nozioni come la relazione, la deviazione, tra due o più item per trovare qualche specifica correlazione tra essi. Nei sistemi di verifica della firma, la firma media (template/pattern) viene elaborata a partire da firme collezionate precedentemente (nella fase di training). Questa viene salvata nella base di conoscenza e quando si ha in input una firma da classificare viene utilizzato il concetto di correlazione per calcolare la distanza tra la firma da verificare e la firma media per poi decidere se accettarla o rifiutarla.

Nella prossima sezione descrivo il classificatore di firme statiche implementato dall'azienda Corvallis prima del progetto di stage.

#### 2.1.4 Prototipo preesistente

Il classificatore di firme statiche, implementato dall'azienda, fa parte della categoria Statistical-based. In particolare è un classificatore di tipo Distance-based.

Il prototipo è in grado di estrarre 21 feature. Il sistema viene addestrato utilizzando un database di firme. Per ciascun sottoscrittore, un vettore contenente il baricentro di ogni feature (centroid feature vector) viene calcolato utilizzando gli esemplari di firma genuini. Il vettore baricentro costituisce il template delle firme del firmatario. In fase di verifica, per misurare la distanza tra la firma template e la firma da testare, viene adoperata la distanza Euclidea nello spazio delle features.

La particolarità di questo sistema sta nel fatto che, per classificare una firma, non si basa soltanto sulla distanza Euclidea ma anche su un meccanismo dei pesi. Si assegna un peso ad ogni feature in



base alla variabilità intrapersonale che essa assume tra i prototipi firma di un sottoscrittore. Questa meccanica è descritta meglio nelle sezioni a seguire.

In figura 8 segue uno schema riassuntivo dei processi interni al classificatore.

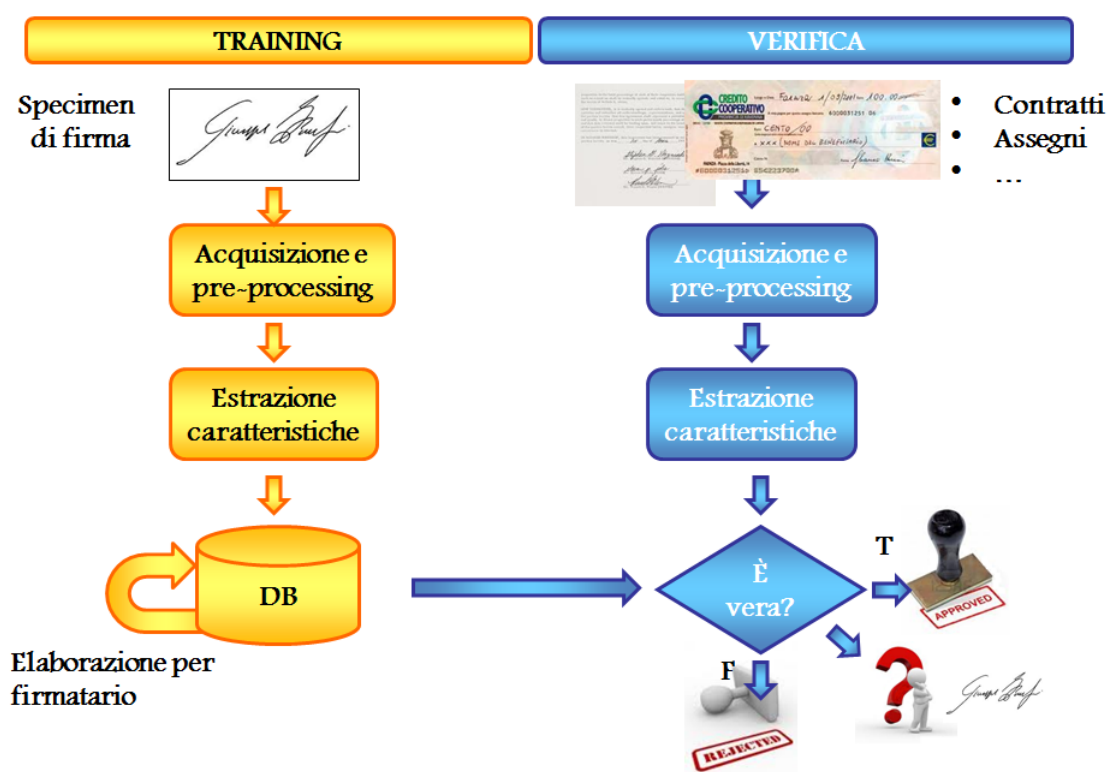


Figura 8: Il processo  
Source:Corvallis

### Fase di training

Il training prende in input N prototipi di firma genuina, una lista di feature da estrarre e i pre-processing necessari ad ogni feature. Restituisce in output un file contenente quattro attributi per feature, utili in fase di verifica:

- average, la media di una feature è data dalla somma dei valori che tale feature assume negli N prototipi diviso N:

$$F_{average} = \frac{\sum_{i=1}^N FValPrototipoI}{N}$$

- *threshold*, la soglia di una feature è data dal massimo valore tra la massima distanza del valore della feature dalla media e il 10% della media:

$$Fthreshold = \max[\max(|FValPrototipoI - Faverage|), (Faverage * 0.1)]$$

- *weight*, il peso di una feature è diverso da utente a utente poiché è inversamente proporzionale alla variabilità della feature rispetto alle altre dell'utente considerato. Il peso di una feature appartiene all'intervallo  $[0, 1]$ .

- il peso vale 0 se la deviazione standard è maggiore del 30% della media;
- il peso vale 1 se la deviazione standard è uguale a 0;
- altrimenti:

$$Fweight = \frac{1}{\log(minimaDeviazioneStandard)} * \log(FdeviazioneStandard)$$

con *minimaDeviazioneStandard* si intende la minima deviazione standard tra tutte le feature considerate; con *FdeviazioneStandard* si intende la deviazione standard per la feature *F*.

- *tollerance*, il valore della tolleranza è inversamente proporzionale alla variabilità delle feature rispetto alle altre (per un utente specifico). Più una feature è poco variabile più il sistema è tollerante nei suoi confronti in fase di verifica. La tolleranza appartiene all'intervallo  $[minTol, maxTol]$ ; *minTol* e *maxTol* sono due dei parametri che possono essere impostati a priori. La tolleranza vale *maxTol* se il peso della feature corrente è massimo (*Fweight*=1). Altrimenti, data la lista ordinata dei pesi, la si può calcolare con la seguente formula:

$$Ftolerance = maxTol - i * \frac{maxTol - minTol}{N - 1}$$

*i* è la posizione *i*-esima nella lista ordinata dei pesi. Poiché il peso e la tolleranza sono direttamente proporzionali essi scalano insieme.

In figura 9 segue uno schema riassuntivo delle operazioni eseguite durante la fase di training.

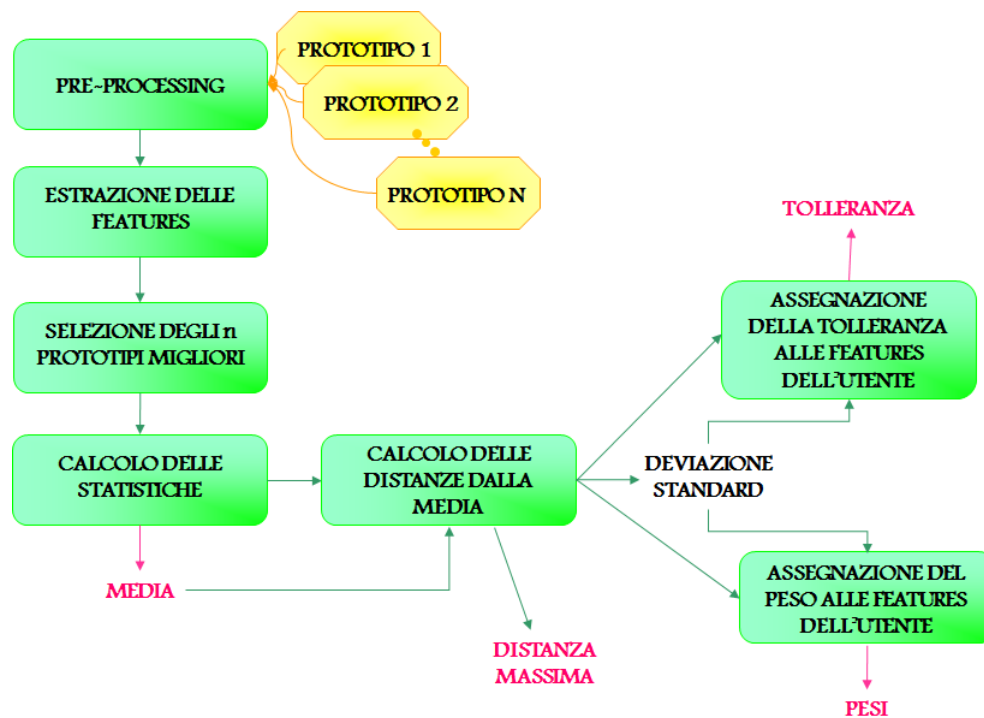


Figura 9: Training per firmatario  
Source: Corvallis

### Fase di verifica

Il processo di verifica prevede i seguenti step:

1. Estrazione delle feature della firma di cui si vuole verificare l'autenticità, secondo la lista feature e preprocessing dell'utente che si vuole personificare;
2. Assegnazione dei voti;
3. Calcolo dei punteggi;
4. Calcolo della confidenza.

### Assegnazione dei voti

Il valore di ogni feature estratta dalla firma da verificare viene confrontato con la media corrispondente (presa dal feature centroid vector). In base a quanto si scosta dalla media, ad ogni feature viene assegnato un voto nell'intervallo  $[-1, 1]$ .

- il voto è uguale a 1 (voto massimo possibile) quando la differenza tra il valore della feature e la relativa media salvata è uguale a 0, ossia il valore della feature è uguale alla media salvata.
- il voto è uguale a -1 (voto minimo possibile) quando la differenza tra il valore della feature e la relativa media salvata è maggiore o uguale a due volte la soglia.

- altrimenti il voto viene assegnato secondo una funzione lineare:

$$Fvoto = \frac{Fthreshold - |FVal - Faverage|}{Fthreshold}$$

### Calcolo dei punteggi

Il core del sistema classifica le firme in base al punteggio che queste riescono a raggiungere. Il punteggio massimo raggiungibile è chiaramente la somma dei pesi delle feature. Vediamo ora come si calcola il singolo punteggio che poi andrà a costituire parte del punteggio finale.

$$Fpunteggio = Fvoto * Fweight$$

Fvoto è il voto assegnato alla feature F e Fweight è il peso della feature F calcolato durante il training.

### Calcolo della confidenza

La decisione finale è data dal valore della confidenza. Se il valore della confidenza è maggiore di 50, la firma viene accettata altrimenti rifiutata. Il calcolo della confidenza è dato da:

$$Confidenza = \frac{0.5 + \sum_{i=1}^M FI_{punteggio}}{2 * \sum_{i=1}^M FI_{weight}} * 100$$

M indica il numero di feature prese in considerazione.

In figura 10 segue uno schema riassuntivo delle operazioni eseguite durante la fase di verifica.

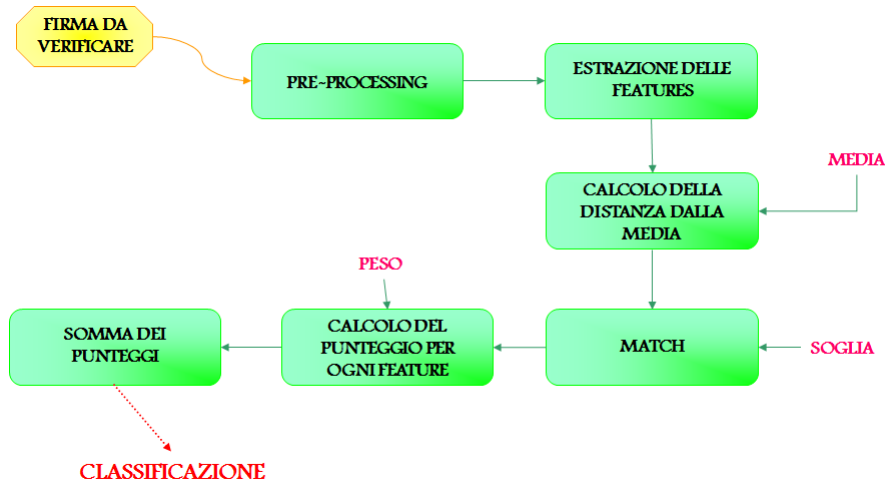


Figura 10: Verifica firma  
Source: Corvallis

### Performance

In tabella 1 vengono riportati i risultati di alcuni test.

FAR random	FAR skilled	FRR
0.10	0.32	0.33
0.05	0.25	0.45
0.00	0.09	0.45
0.00	0.14	0.42

Tabella 1: Risultati test prototipo preesistente

### Tecnologie impiegate

Il prototipo è stato scritto in Java, utilizzando l'ide Eclipse. Le librerie utilizzate per l'elaborazione delle immagini sono ImageJ e ImageMagick. Il versionamento del codice è stato effettuato grazie a Content Versioning System (CVS).

La prossima sezione introduce il progetto di stage a cui ho preso parte.

## 2.2 Obiettivi

L'obiettivo principale dello stage è lo studio e l'implementazione di un nuovo classificatore di firme statiche, sempre nell'ambito dei classificatori Statistical-based. Uno degli obiettivi secondari è testare approfonditamente il software preesistente per identificare le eventuali cause di un bug che rende la classificazione attuale dipendente dalla risoluzione del dispositivo di cattura. L'obiettivo ultimo è quello di valutare le performance del nuovo classificatore e di studiare una modalità di unione delle classificazioni dei due prototipi in modo da aumentare l'accuratezza globale del sistema di verifica delle firme statiche.

## 2.3 Aspettative

L'azienda si aspetta che l'obiettivo principale venga raggiunto.

## 2.4 Vincoli

Non sono stati imposti vincoli particolari per quanto riguarda lo svolgimento del progetto di stage. I vincoli imposti sono:

- utilizzare il linguaggio Java per l'implementazione del prototipo
- fornire documentazione adeguata sul prototipo implementato
- utilizzare le librerie ImageJ e/o ImageMagick per l'elaborazione delle immagini
- lo studio di nuovi metodi di classificazione deve avvenire sempre nell'ambito dei classificatori Statistical-based

## 3 Attività di stage

### 3.1 Pianificazione

### 3.2 Analisi

#### 3.2.1 Ricerca di migliorie del prototipo preesistente

#### 3.2.2 Scelta di un nuovo classificatore di firme statiche

#### 3.2.3 Requisiti

### 3.3 Progettazione

### 3.4 Implementazione

## 4 Valutazione retrospettiva

### 4.1 Copertura dei requisiti

#### 4.1.1 Possibili sviluppi alle attività svolte

### 4.2 Conoscenze acquisite

### 4.3 Distanza tra conoscenze richieste e conoscenze possedute

## Glossario

**dematerializzazione** processo che ha come obiettivo ultimo la creazione di un flusso di documenti digitali aventi pieno valore giuridico, che vada prima ad affiancare e poi, sul lungo periodo, a sostituire la normale documentazione cartacea presente negli archivi di qualunque attività pubblica o privata.. [8](#)

**SOA** un'architettura software adatta a supportare l'uso di servizi Web per garantire l'interoperabilità tra diversi sistemi così da consentire l'utilizzo delle singole applicazioni come componenti del processo di business e soddisfare le richieste degli utenti in modo integrato e trasparente.. [6](#)



## Riferimenti bibliografici

- [1] Hemanta Saikia, Kanak Chandra Sarma *Approaches and issues in Offline Signature Verification System.*
- [2] Battista L., Rivard D., Sabourin R., Granger E., Maupin P., *State of the art in off-line signature verification.*
- [3] Justino, Yacoubi, Bortolozzi, Sabourin *An Off-Line Signature Verification System Using HMM and Graphometric Features.*
- [4] Yazan M. Al-Omari, Khairuddin Omar *State-of-the-Art in Offline Signature Verification System.*