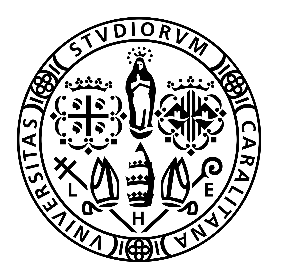
Università degli Studi di Cagliari



Dipartimento di Matematica e Informatica

Laurea Triennale in Informatica Applicata

e Data Analytics

Stato dell'arte sulle tecniche di data augmentation nella rilevazione di falsi di impronta digitale

|  |  |
| --- | --- |
| *Relatore*  Dr. Marco Micheletto  *Correlatrice*  Dr.ssa Giulia Orrù | *Candidato*  Gian Maria Alvau  Matricola 60/79/00053 |
| Anno Accademico 2024/2025 | |

*Ai miei genitori e a mia sorella,*

*che mi hanno sempre supportato.*

*A Maria,*

*per essere sempre stata al mio fianco.*

Indice

[Abstract 4](#_Toc182257832)

[Capitolo 1 5](#_Toc182257833)

[Introduzione 5](#_Toc182257834)

[1.1 Biometria 5](#_Toc182257835)

[1.2 Impronte digitali 6](#_Toc182257836)

[1.2.1 Acquisizione di impronte 8](#_Toc182257837)

[1.3 Tipi di riconoscimento 10](#_Toc182257838)

[1.4 Architettura generale di un sistema di riconoscimento 11](#_Toc182257839)

[1.5 Introduzione agli attacchi e ai rischi 12](#_Toc182257840)

[Capitolo 2 15](#_Toc182257841)

[Attacchi di presentazione e sicurezza nei sistemi biometrici 15](#_Toc182257842)

[2.1 Falsificazione di impronte 15](#_Toc182257843)

[2.2 Presentation Attack Detection (PAD) 16](#_Toc182257844)

[2.3 Metodi di rilevazione allo stato dell’arte 17](#_Toc182257845)

[2.4 Metodi di valutazione 18](#_Toc182257846)

[2.5 Dataset 19](#_Toc182257847)

[2.6 Limiti degli attuali sistemi PAD 20](#_Toc182257848)

[Capitolo 3 23](#_Toc182257849)

[Data Augmentation 23](#_Toc182257850)

[3.1 Tecniche di Data Augmentation 23](#_Toc182257851)

[3.2 Data Agumentation nelle immagini 23](#_Toc182257852)

[3.2.1 Augmentation di immagini singole sul valore 24](#_Toc182257853)

[3.2.2 Augmentation di immagini singole sulla struttura 25](#_Toc182257854)

[3.2.3 Augmentation di immagini singole su valore e struttura 27](#_Toc182257855)

[3.2.4 Augmentation di immagini multiple sul valore 28](#_Toc182257856)

[3.2.5 Augmentation di immagini multiple basate sulla struttura 30](#_Toc182257857)

[3.2.6 Augmentation di immagini basate sulla popolazione 30](#_Toc182257858)

[3.3 Applicazioni su FPAD 31](#_Toc182257859)

[3.4 Limiti della Data Augmentation 31](#_Toc182257860)

[Capitolo 4 33](#_Toc182257861)

[Generazione sintetica per l’addestramento di rilevatori di attacchi di presentazione 33](#_Toc182257862)

[4.1 Modelli GAN per la genereazione sintetica 33](#_Toc182257863)

[4.1.1 SpoofGAN 34](#_Toc182257864)

[4.1.2 FPGAN-Control 36](#_Toc182257865)

[4.1.3 Universal Material Generator 37](#_Toc182257866)

[4.2 Modelli probabilistici di diffusione 41](#_Toc182257867)

[4.2.1 GenPrint 42](#_Toc182257868)

[4.4 Data Augmentation e generazione sintetica a confronto 44](#_Toc182257869)

[Capitolo 5 46](#_Toc182257870)

[Conclusioni 46](#_Toc182257871)

[Bibliografia 47](#_Toc182257872)

# Abstract

L'uso delle impronte digitali come metodo di identificazione è uno dei più diffusi e consolidati nel campo della biometria, grazie alla loro unicità e affidabilità. I progressi tecnologici hanno reso i sistemi di identificazione automatici sempre più rapidi, precisi e sicuri, trovando applicazione in ambiti come banche, documenti di identità, smartphone e altri dispositivi. Tuttavia, con l’evoluzione di questi sistemi, sono emerse tecniche volte a eluderli, tra cui gli attacchi di presentazione, in cui vengono utilizzate impronte falsificate per ingannare i sensori biometrici.

Per affrontare questa sfida, i recenti sviluppi nell'ambito del Deep Learning hanno introdotto algoritmi in grado di rilevare con maggiore precisione gli attacchi di presentazione rispetto ai metodi tradizionali basati su caratteristiche manuali. Tuttavia, tali approcci richiedono grandi quantità di dati per l'addestramento, spesso difficili da reperire, limitando così l'efficacia e la generalizzazione dei modelli in contesti reali.

Questa tesi affronta il problema della scarsità di dati esaminando l'uso della data augmentation e della generazione sintetica di impronte digitali come soluzioni per potenziare i sistemi di rilevamento degli attacchi di presentazione, analizzando le modalità con cui queste tecniche possono contribuire a migliorare la robustezza dei sistemi biometrici e supportare lo sviluppo di metodi più efficaci nel rilevamento di attacchi mai visti prima.

# Capitolo 1

# Introduzione

La biometria è la disciplina scientifica che si occupa della misurazione delle grandezze biofisiche dell’individuo e del loro utilizzo per l’identificazione personale. I sistemi utilizzati per il riconoscimento dell’individuo tramite queste grandezze sono chiamati sistemi biometrici e sono oggi altamente diffusi come sistemi di sicurezza. I sistemi biometrici non si basano su ciò che la persona conosce, come una password, ma su ciò che la persona possiede, quindi caratteristiche come volto, iride, impronte digitali o del suo comportamento per l’identificazione dell’individuo. Non è possibile copiare un identificatore biometrico nello stesso modo con cui si copierebbe una password. I sistemi biometrici non sono inviolabili, sono soggetti a vari tipi di attacchi che possono agire in diverse fasi del sistema, influenzando i dati o il metodo di funzionamento di questo. In questa tesi verrà analizzato il problema degli attacchi di presentazione (presentation attack) nei sistemi biometrici basati sulle impronte digitali. Le impronte digitali sono uno degli identificatori biometrici più usati in vari settori, tra cui il settore bancario e le istituzioni pubbliche. Gli attacchi di presentazione consistono nella presentazione al sensore di un carattere biometrico falsificato per eludere l’identificazione; nel caso delle impronte digitali si tratta di impronte false che possono essere create con vari materiali, tra cui silicone, lattice e gomma. Essendo una tipologia di attacco che non necessita di specifiche capacità tecniche, ma solo della tecnica di contraffazione, il numero di potenziali aggressori è molto alto, così come lo sono i materiali che possono essere utilizzati per ricreare un’impronta. A causa di questi fattori risulta necessario lo sviluppo di sistemi sofisticati per la rilevazione degli attacchi di presentazione, che prendono il nome di sistemi di presentation attack detection (PAD).

## Biometria

Con il termine biometria si identifica sia la disciplina scientifica che le grandezze da essa misurate, anche note come tratti biometrici. I tratti biometrici, o biometrie, sono proprietà biofisiche del corpo umano e vengono suddivise in biometrie fisiologiche e comportamentali. Le biometrie fisiologiche sono caratteri fisicamente posseduti da un individuo, come viso, iride, impronta digitale, impronta della mano o lo schema delle vene del palmo; mentre le biometrie comportamentali si basano su una misurazione nel tempo di un comportamento dell’individuo come l’andatura della camminata, l’elettrocardiogramma, l’encefalogramma o la sua voce [80]. Le biometrie comportamentali risultano meno invasive e più dinamiche, adattandosi alla variazione del comportamento dell’individuo, ma richiedono un tempo più lungo per la misurazione e l’apprendimento del carattere biometrico [81]. Le biometrie fisiche risultano più stabili nel tempo e difficili da falsificare, ma sono più invasive. Ogni carattere biometrico ha delle proprietà che lo rendono differente dagli altri (Tabella 1.1), permettendo di sviluppare sistemi biometrici su misura per il caso d’uso. Le proprietà fondamentali per un qualsiasi carattere biometrico [3] sono:

* Universalità, ogni individuo deve possederlo.
* Unicità, deve essere diverso per ogni individuo.
* Permanenza, deve essere invariabile nel tempo.
* Collezionabilità, deve poter essere acquisito.

Tabella 1.1 Proprietà dei caratteri biometrici, in cui A significa Alta/o, M significa Media/o e B significa Bassa/o. Sono mostrati i valori delle caratteristiche per i caratteri biometrici più utilizzati, sia fisiologici che comportamentali.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Universalità** | **Unicità** | **Permanenza** | **Collezionabilità** | **Prestazioni** | **Accettabilità** | **Robustezza** | **Costo** | **Maturità** |
| **Impronte Digitali** | M | A | A | M | A | M | B | M | A |
| **Volto** | A | B | M | A | B | A | A | B | M |
| **Iride** | A | A | A | M | A | B | B | A | M |
| **Vene del Palmo** | M | M | M | M | M | M | A | A | B |
| **Termogramma Facciale** | A | A | B | A | M | A | A | A | M |
| **Retina** | A | A | M | B | A | B | A | A | A |
| **Orecchio** | M | M | A | M | M | A | M | M | B |
| **Geometria della Mano** | M | M | M | A | M | M | M | B | A |
| **Voce** | B | B | B | A | B | A | B | M | M |
| **Firma** | B | B | B | A | B | A | B | M | M |
| **Andatura** | M | B | B | A | B | A | M | A | B |

## 1.2 Impronte digitali

Le impronte digitali sono una delle biometrie più utilizzate grazie alla loro immutabilità, individualità, collezionabilità e facilità di confronto. I primi casi di utilizzo delle impronte digitali per l’identificazione di un individuo risalgono ai contratti scritti nelle steli di argilla dell’antica Babilonia, in cui venivano usate come firma, e sono state utilizzate ampiamente dalle forze di polizia per l’identificazione di criminali a partire dalla fine del 1800 [93]. Grazie alla loro diffusione, le impronte digitali sono state studiate approfonditamente [82].

Un’impronta digitale è una traccia lasciata dal dermatoglifo dell’ultima falange delle dita delle mani. Il dermatoglifo è formato da un’alternarsi di pieghe della pelle, chiamate creste e valli, che si forma nel feto durante l’ottavo mese di gravidanza. Una cresta è un segmento curvo, mentre una valle è composta dall’area tra due diverse creste.

Quando il dermatoglifo viene danneggiato da tagli o graffi, questo si ricostituisce con la stessa medesima forma, a prescindere dall’età dell’individuo. L’individualità è provata empiricamente poiché, nonostante matematicamente non sia mai stato provata l’unicità di ogni impronta, non sono ancora state trovate impronte digitali completamente identiche, neanche nel caso di gemelli omozigoti.

Le caratteristiche distintive delle impronte digitali sono create dal particolare pattern creato dalle creste e sono suddivise in 3 livelli: L1, L2 ed L3 [84]. Le feature L1 sono visibili facilmente ad occhio nudo e sono create dal particolare schema delle creste in specifici punti, chiamati core e delta, che appaiono sempre in coppia. In base alla forma di core e delta viene definita la classe dell’impronta digitale, di cui esistono 5 tipologie: arch, tended arch, left loop, right loop e whorl (Figura 1.1).

Le feature L2 sono chiamate minutiae e sono il fattore più discriminante tra due impronte digitali visto il numero e le diverse tipologie con cui si presentano.

I tipi di minutiae sono:

* Le terminazioni delle creste;
* Le biforcazioni, punti in cui una cresta si suddivide in due o più creste;
* Le creste indipendenti;
* I punti e le isole, creste di dimensioni molto ridotte;
* Laghi, valli tra due creste divergenti;
* Larkspur;
* Attraversamenti, punti in cui si connettono due creste.

Immagine che contiene lastra dei raggi X

Descrizione generata automaticamente

#### Figura 1.1: Le 5 classi di impronte digitali, in cui i core sono evidenziati da cerchi e i delta sono evidenziati da triangoli [1].

I sistemi biometrici per le impronte solitamente si basano sullo schema descritto dalle creste, che risulta sufficientemente descrittivo per un’impronta [1, 76, 77, 78].

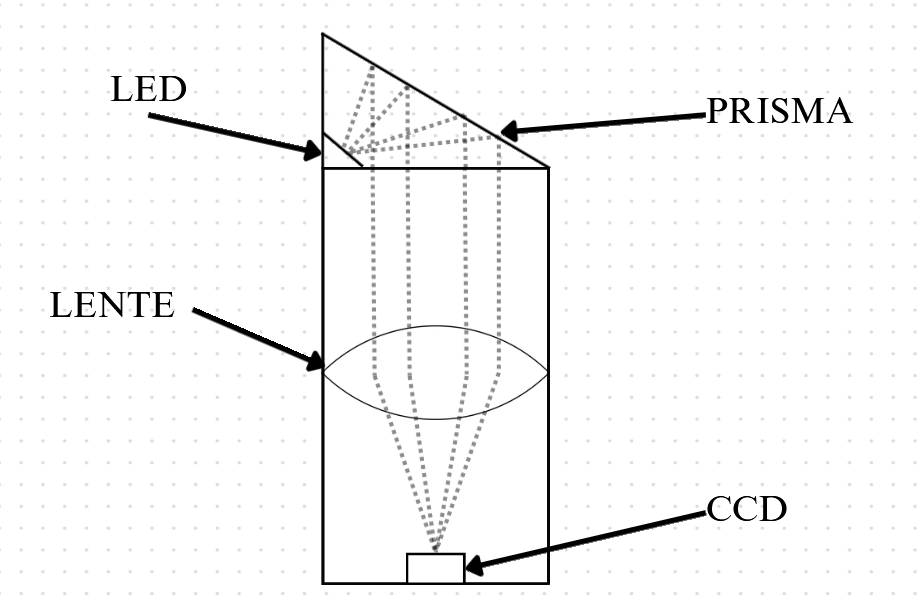
Le feature L3 sono minuscole differenze difficilmente visibili ad occhio nudo, come i pori, le pieghe ed i graffi della pelle, che storicamente sono state meno utilizzate nei sistemi biometrici a causa della dipendenza dalla qualità della rilevazione.

### 1.2.1 Acquisizione di impronte

La prima metodologia per l’acquisizione di impronte digitali è stata la rilevazione tramite inchiostro, che è stata superata in seguito al progresso tecnologico dai metodi basati su sensori.

I sensori per il rilevamento delle impronte digitali sono di svariati tipi, ognuno con una diversa sensibilità e resistenza a fattori che possono comprometterne la corretta rilevazione, come la sporcizia o l’umidità. Storicamente i più utilizzati sono stati i sensori ottici, di cui esistono varie tipologie (CCD, CMOS, CIS).

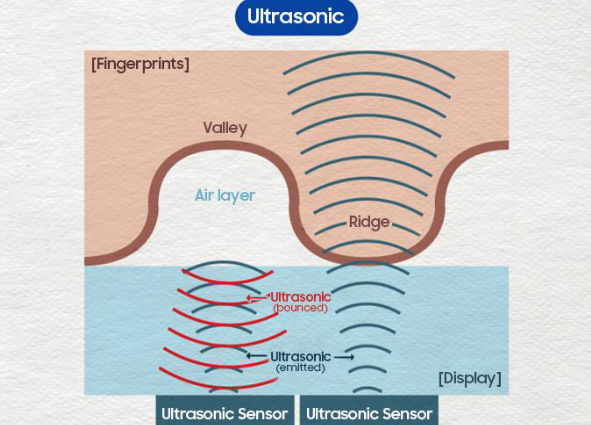
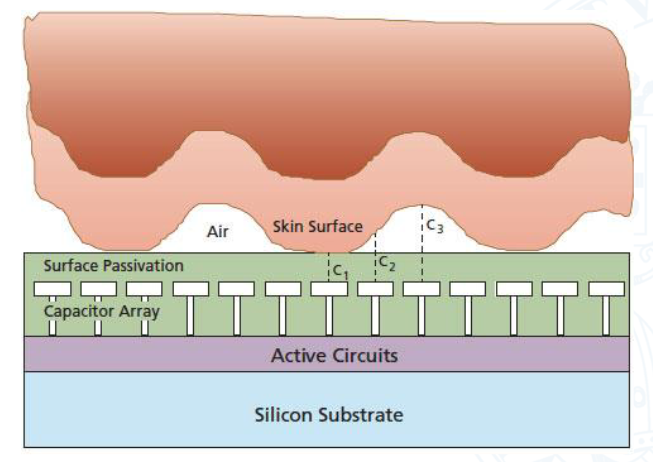
I primi sensori ottici (Figura 1.2) erano di tipo CCD e CMOS; le due tecnologie vennero sviluppate contemporaneamente. La tecnologia CMOS necessitava di una potenza di calcolo non presente in quel periodo, perciò le rilevazioni risultavano a bassa risoluzione, con molto rumore, con bassa sensibilità alla luce e di scadente qualità. Per questi motivi la tecnologia CCD ha dominato il mercato per decenni grazie alla ampia sensibilità alla luce, acquisizione uniforme, basso rumore e altri vantaggi. Con l’innovazione nei sistemi integrati, i problemi della tecnologia CMOS sono stati superati: le minori dimensioni dei sensori, il basso consumo di energia e la possibilità di integrazione di nuove tecnologie hanno permesso ai sensori CMOS di rimpiazzare i sensori CCD. In tempi recenti grandi aziende come Omni Sonic e Toshiba hanno portato grandi sviluppi nei pixel CIS, introducendo un nuovo trend. Nuove tecnologie, come la optical Thin Film Transistor (TFT) [30], continuano ad ampliare il settore, migliorando la qualità delle rilevazioni di impronte [27].



#### Figura 1.2 Schema del funzionamento di un sensore ottico. Appoggiando il dito sul prisma viene attivato il led, che proietta la luce. Questa riflette sul prisma e la lente la converge sul sensore CCD.

Con lo sviluppo dei semiconduttori, sono stati creati vari sensori capacitivi per la rilevazione di impronte. Generalmente una piastra conduttiva entra in contatto con il dermoglifo che, a causa delle creste e delle valli, non tocca in egual modo tutta la piastra. I sensori capacitivi sottostanti la piastra registrano la variazione di corrente nei vari punti di contatto e la registrano sotto forma di dati. Un sensore ottico sotto il display utilizza la differenza di luce riflessa dalle valli e dai solchi per riconoscere l’impronta che tocca la cover sopra il sensore, ma hanno difficoltà a riconoscere dita asciutte, che non creano un contatto regolare e consistente con la cover del sensore. Simile è il funzionamento dei sistemi a ultrasuoni [28], che mostrano molta possibilità di sviluppo, o dei sensori termici [29]. In generale i sensori capacitivi necessitano un grande miglioramento nell’individuazione dell’impronta per essere competitivi, date le scarse performance in caso di sporcizia e umidità.

I sensori più precisi e accurati per acquisire immagini di impronte digitali al giorno d’oggi sono i sensori ad ultrasuoni [27] (Figura 3). Questi sensori sono basati sulla riflessione di ultrasuoni che si propagano in materiali di diversa impedenza. Quando il dito viene appoggiato sul sensore di contatto, questo attiva il sensore ultrasonico. A causa della diversa impedenza tra l’aria e la pelle umana, l’ampiezza dell’echo della pelle umana è maggiore di quella dell’aria, viene così creato un pattern che permette di rilevare l’impronta. Grazie all’alta penetratività degli ultrasuoni, l’impronta può essere rilevata attraverso una piccola quantità di sporcizia o umidità, penetrando anche materiali come vetro, plastica, alluminio e acciaio.

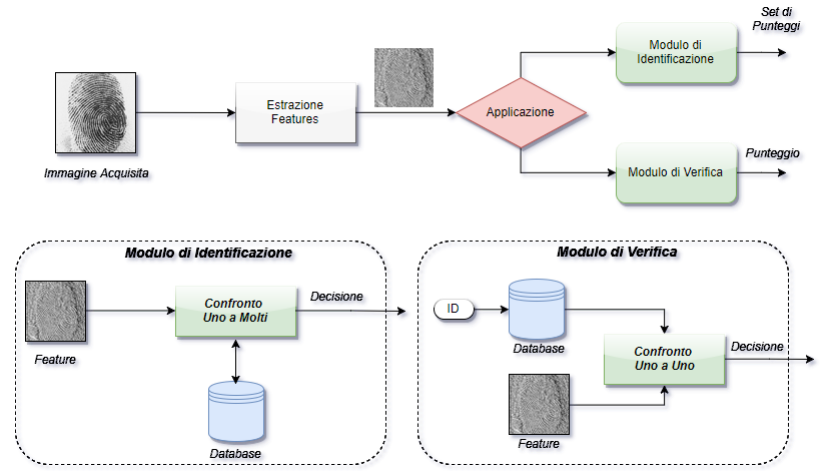


#### Figura 1.3 Funzionamento di un sensore capacitivo (a sinistra) e di un sensore ad ultrasuoni (a destra) per impronte digitali.

## 1.3 Tipi di riconoscimento

I sistemi biometrici possono analizzare una sola tipologia di biometrie (iride, volto, voce, ECG, ecc.), in questo caso sono chiamati sistemi unimodali, oppure una combinazione di biometrie diverse per una maggiore accuratezza e sicurezza, in questo caso sono chiamati sistemi multimodali [31]. I sistemi multimodali comportano varie sfide, tra cui la gestione di dati diversi, la scalabilità e l’intrusività del sistema. Tuttavia i sistemi multimodali permettono di differenziare molto più efficacemente gli utenti data la dimensionalità dei vettori di caratteristiche raccolti, rendendolo allo stesso tempo più resiliente a vari tipi di attacchi. In questa tesi saranno esaminati i sistemi unimodali per le impronte digitali.

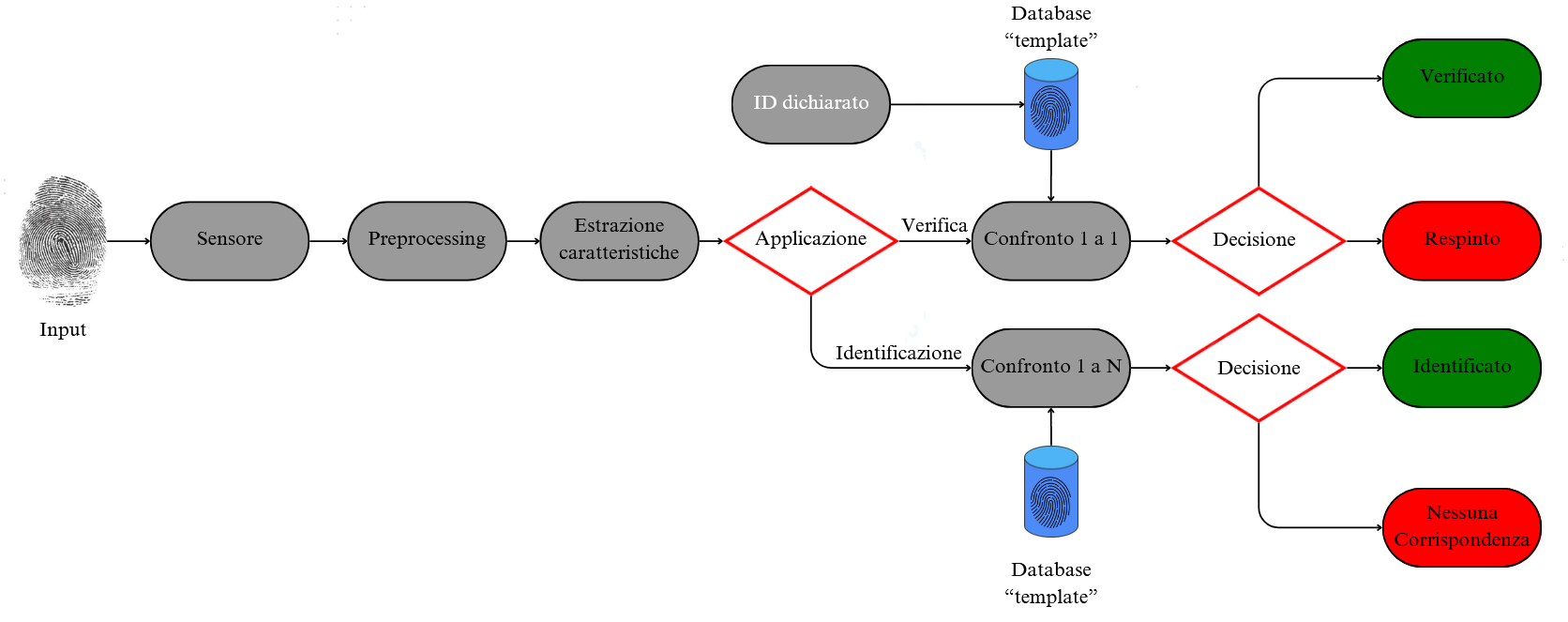
Un sistema biometrico può essere progettato per due tipi di riconoscimento, detti verifica e identificazione, che vengono utilizzati per casi d’uso differenti [83]. L’identificazione viene fatta tramite il confronto dei dati biometrici della persona con un database di modelli biometrici, verificando che i dati della persona siano presenti per consentire l’accesso; un confronto del genere è detto “uno-a-molti” o “uno-a-N”. L’identificazione è tipicamente utilizzata per consentire l’accesso ad un gruppo di persone ad un luogo tramite badge o password, oppure per la ricerca di sospettati in un database. Una verifica invece è un confronto “uno-a-uno” con l’obiettivo di confermare l’identità dichiarata, ovvero un controllo in cui i dati biometrici rilevati sono confrontati con l’unico modello a cui ha accesso il sistema, ovvero il modello del proprietario. La verifica risulta un’operazione più semplice dell’identificazione, visto che abbiamo un solo controllo, mentre l’identificazione comprende molti controlli. Le applicazioni più comuni di questi sistemi sono lo sblocco del telefono e il login a sistemi personali.



#### Figura 1.4 Schema del funzionamento di un sistema di riconoscimento in cui viene mostrata la differenza tra identificazione e verifica.

## 1.4 Architettura generale di un sistema di riconoscimento

In un sistema di riconoscimento biometrico (Figura 1.5) il processo di riconoscimento inizia con la rilevazione dell’impronta da parte del sensore. L’immagine ottenuta dal sensore viene manipolata per migliorarne la qualità e facilitare l’estrazione delle caratteristiche. Solitamente viene prima segmentata per centrare l’immagine sull’impronta e successivamente viene applicata la binarizzazione (operazione con la quale i valori dei pixel vengono trasformati in 0 o 1 in base ad una soglia). Infine le creste vengono assottigliate [39] e viene effettuata l’estrazione delle caratteristiche con algoritmi specializzati o con reti neurali. In base all’applicazione specifica viene fatto un confronto 1-a-N con i modelli del database (identificazione) oppure un confronto 1-a-1 tra il modello derivato dall’impronta fornita dall’utente ed i modelli nel database dell’utente dichiarato. In entrambi i casi il sistema fornisce un valore di output rappresentante la similarità tra i modelli che, se superiore alla soglia di similarità del sistema decisa a priori, consentirà l’accesso all’utente.



#### Figura 1.5. Schema completo di un sistema di riconoscimento di impronte digitali.

## 1.5 Introduzione agli attacchi e ai rischi



#### Figura 1.6 Schema semplificato del funzionamento di un sistema biometrico, con alcuni dei vari tipi di attacchi che agiscono in diversi punti del processo di comparazione.

Durante il processo di analisi delle impronte i sistemi possono essere vulnerabili a diversi tipi di attacchi [84], mostrati nella Figura 1.6. In questa tesi saranno analizzati nello specifico i Presentation Attack (PA) [7, 12], ovvero la sostituzione dell’impronta reale con una finta, detta anche fake o spoof, creata utilizzando diversi materiali tra cui lattice, gelatina, colla vinilica ed altri. I Presentation Attack rappresentano una seria problematica per i sistemi di riconoscimento, poiché il numero di potenziali aggressori e variazioni nell’attacco è molto alto.

I PA, infatti, rappresentano una problematica reale per molti settori, essendo le impronte digitali utilizzate ampiamente per l’identificazione di un individuo da telefoni, banche e istituzioni pubbliche.

Altri tipi di attacchi sono:

* La replica del segnale biometrico (2), che agisce tra l’invio dei dati dal sensore e le operazioni di preprocessing che consiste nel replicare un segnale precedentemente salvato, come una vecchia copia del segnale di un’impronta digitale;
* La modifica delle feature (3), che è compiuta con un attacco Cavallo di Troia, con cui si immette un set di caratteristiche impostore nel sistema;
* La sostituzione delle feature (4) consiste nella sostituzione del set di caratteristiche prima che questo raggiunga il comparatore;
* L’elusione del comparatore (5) viene effettuata scambiando i punteggi di comparazione con altri punteggi;
* La sostituzione del template (6) è la manomissione dei template nel database, che può permettere l’autenticazione di un’identità non presente o vietare l’autenticazione di un utente genuino;
* L’intercettazione della comunicazione (7) agisce sul canale di comunicazione tra database e comparatore, variando i dati durante il trasporto;
* L’alterazione della decisione finale (8) comprende gli attacchi con cui è possibile variare o bloccare la decisione del sistema [9].

Fortunatamente, con lo sviluppo tecnologico, sono state implementate diverse tecniche per la rilevazione e prevenzione degli attacchi di presentazione, che prendono il nome di rilevazione di attacchi di presentazione (PAD, Presentation Attack Detection). I sistemi PAD odierni utilizzano algoritmi di apprendimento automatico (Machine Learning) per analizzare le impronte digitali e decidere se la rilevazione del sensore appartenga ad un’impronta vera o falsificata, inferendo la decisione in base ad un dataset con cui vengono addestrati [5, 6]. Le prestazioni di questi algoritmi sono fortemente dipendenti dal dataset, in cui devono essere presenti molti esempi di impronte vere e false per apprendere efficacemente le differenze. Inserire un PAD in un sistema di riconoscimento può inficiare sulle sue performance, perciò vi è il bisogno di trovare un equilibrio tra sicurezza e usabilità in base all’utilizzo del sistema [86]. Un sistema PAD è migliore quanto maggiore è la sua capacità di generalizzazione, ossia la capacità di rilevare un attacco anche quando presenta caratteristiche diverse da quelle viste in fase di addestramento, come il materiale dell’impronta o l’angolazione. Tuttavia, per raggiungere questo livello di robustezza, è necessario disporre di un dataset sufficientemente esteso e rappresentativo.

La creazione di un dataset con queste caratteristiche richiede la raccolta di una grande quantità di impronte vere, per cui necessita un alto numero di utenti consenzienti; inoltre occorrono anche molte impronte falsificate, che richiede conoscenze, materiali, strumentazione e tempo. Questi fattori, insieme alla maggiore difficoltà di raccolta e trattamento dei dati sensibili, rendono l’ottenimento di dataset ottimali molto difficile.

Data la scarsa disponibilità di dati sono state sviluppate tecniche per creare nuovi campioni a partire da quelli già presenti, dette tecniche di Data Augmentation [87], o nuovi campioni sintetici da immettere nel dataset, chiamate tecniche di generazione sintetica.

Questa tesi si pone come obiettivo l’analisi dei sistemi odierni allo stato dell’arte per la Data Augmentation e la generazione sintetica di impronte digitali falsificate con l’utilizzo di algoritmi di Intelligenza Artificiale, atti a migliorare le performance dei sistemi di rilevazione di attacchi di presentazione.

Il resto della tesi è strutturato come segue: Il Capitolo 2 tratta la falsificazione di impronte e i sistemi di Presentation Attack Detection allo stato dell’arte. Nel Capitolo 3 viene definito cosa sia la Data Augmentation e le principali tecniche utilizzate nel Machine Learning, per vederne le applicazioni nella Fingerprint Presentation Attack Detection (FPAD). Il Capitolo 4 affronta la generazione sintetica di impronte digitali e le principali metodologie di generazione basate su modelli di Deep Learning, L’ultimo Capitolo è il capitolo conclusivo in cui vengono mostrate le conclusioni del seguente lavoro di tesi ed eventuali lavori futuri per perfezionare i meccanismi di Data Augmentation e generazione sintetica per i sistemi FPAD.

# Capitolo 2

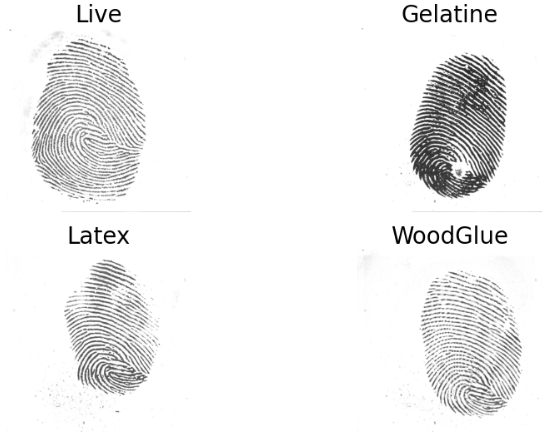
# Attacchi di presentazione e sicurezza nei sistemi biometrici

In questo Capitolo verrà introdotto il problema della riproduzione di impronte digitali ed i sistemi creati per risolvere questo problema, ovvero i sistemi di Presentation Attack Detection. I PAD si sono evoluti nel tempo, passando da algoritmi definiti hand-crafted, ad indicare il fatto che si basassero su particolari modelli con metriche definite a priori, ad algoritmi basati su una branca specifica dell’intelligenza artificiale, il Deep Learning [6,19,20,21]. Questi algoritmi sono in grado di imparare autonomamente le caratteristiche distintive degli oggetti che vengono forniti in addestramento e sono in grado di fornire in tempi brevissimi una decisione sull’autenticità dell’impronta, con una precisione maggiore dei metodi tradizionali. Tuttavia non sono ancora perfetti, soprattutto per la loro dipendenza dai dati forniti in addestramento che devono essere forniti in gran numero e manipolati correttamente per ottimizzare il processo di addestramento.

## 2.1 Falsificazione di impronte

Il processo di creazione di impronte false [7, 12] richiede l’ottenimento dell’impronta di un individuo, che può essere consenziente o meno. Nel caso in cui l’individuo sia consenziente, premendo il dito su di un materiale plastico potrà essere creato uno stampo dell’impronta che consentirà riproduzioni ad alta fedeltà, il caso peggiore per un sistema di Presentation Attack Detection. I metodi consensuali permettono la migliore qualità possibile di impronte falsificate, ma non sono casi realistici di attacchi, visto che la vittima stessa dovrebbe dare il suo consenso.

I metodi non consensuali per la creazione di impronte false rappresentano il caso più reale in cui un aggressore tenti di eludere il sistema di sicurezza per ottenere accesso ai dati appartenenti ad un altro individuo senza la sua collaborazione o consenso. In questi casi si cerca di ottenere un’impronta digitale latente da una superficie, solitamente liscia, come lo schermo di un telefono, per digitalizzarla e riprodurla. Per evidenziare e acquisire le impronte latenti vengono utilizzati vari tipi di polveri [8], oppure si possono ottenere delle immagini da cui poi viene creato lo stampo [4]. Acquisizioni di questo tipo risultano meno fedeli e dettagliate, data la difficoltà nell’ottenere l’intero schema delle creste di un’impronta senza distorsioni o sporcizia, rispetto agli attacchi consensuali.



#### Figura 2.1 Rilevazione di un’impronta digitale vera (Live) e tre versioni falsificate create con materiali diversi: gelatina, lattice e colla per legno.

## 2.2 Presentation Attack Detection (PAD)

Con Presentation Attack si intendono quegli attacchi ai sistemi biometrici che mirano all’immissione nel sistema di un’impronta fittizia tramite il sensore di rilevamento, con lo scopo di avere accesso a dati di un altro individuo senza il suo consenso. Un sistema di Fingerprint Presentation Attack Detection (FPAD) (Figura 2.1) è un sistema hardware o software utilizzato per stabilire se un’impronta sia “Live” (vera) o “Fake” (falsa), e quindi se l’impronta acquisita del sensore appartenga ad un individuo reale o sia stata falsificata. I sistemi FPAD possono essere suddivisi in due categorie generali: basati sull’hardware (hardware-based), se utilizzano specifici sensori inseriti nel dispositivo per la rilevazione con lo scopo di individuare delle proprietà fisiologiche delle impronte, oppure basati sul software (software-based) se utilizzano sistemi per la rilevazione delle proprietà dell’impronta dopo l’acquisizione [10]. I sistemi hardware-based hanno dimostrato una maggiore percentuale di rilevamento di immagini false, ma i modelli software-based sono meno costosi e intrusivi; la soluzione ottimale sarebbe un PAD che utilizza entrambi gli approcci [10]. In questa tesi non verranno esaminate le tecniche basate sull’hardware.

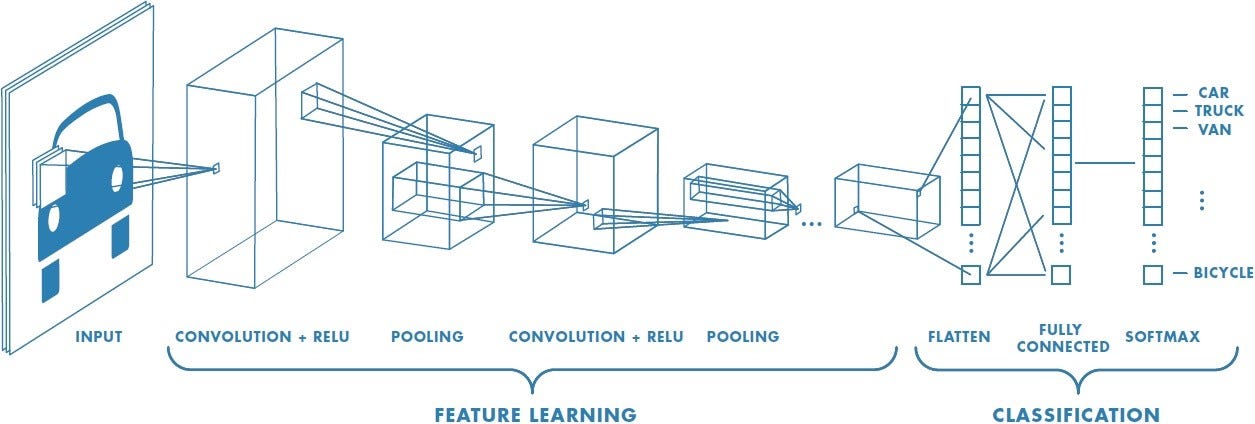
I modelli basati sul software sono basati su paradigmi che analizzano diverse caratteristiche della rilevazione, per esempio gli algoritmi tessiturali utilizzano il pattern delle creste e delle valli insieme a caratteristiche della pelle come la presenza di pori e la perspirazione per differenziare le impronte. I metodi dinamici, invece, richiedono varie rilevazioni per le quali si misurano distorsioni o cambiamenti temporali [17]. Le rilevazioni vengono prima binarizzate, poi “scheletrizzate” (assottigliate) e possono poi essere utilizzati filtri, ad esempio i filtri di Gabor, per migliorare la qualità dell’immagine e facilitarne l’estrazione delle feature [16].



#### Figura 2.2 Schema di funzionamento di un PAD.

## 2.3 Metodi di rilevazione allo stato dell’arte

I sistemi di Presentation Attack Detection si sono evoluti insieme alle tecniche di Machine Learning. I primi sistemi utilizzavano algoritmi tessiturali (Local Binary Pattern, BSIF) per estrarre le caratteristiche delle impronte con un classificatore semplice [16], ma, con l’avvento delle tecniche di Deep Learning, le reti convoluzionali sono diventate il nuovo stato dell’arte. Gli algoritmi tessiturali codificano la struttura di un’impronta in un vettore di caratteristiche utilizzando set di filtri. I metodi più utilizzati a questo scopo sono LBP [33], LPQ [34] e BSIF [32]. Con lo sviluppo delle reti neurali si è trovato un nuovo paradigma su cui basare i sistemi PAD: i modelli di Machine Learning di tipo Deep. I modelli Deep sono reti stratificate di tipo feed-forward (ogni strato passa il suo output solo agli strati sottostanti) che utilizzano l’algoritmo di retropropagazione per variare la matrice dei pesi e diminuire la funzione di perdita in base alla discesa del gradiente [14]. La funzione di perdita è la funzione con cui si calcola la differenza tra output atteso e output reale della rete. Le reti convolutive sono basate su una serie di livelli convoluzionali e di pooling, che termina con un livello completamente connesso finale. In una convoluzione, una maschera di dimensioni variabili viene apposta in vari settori di un’immagine e produce una mappa delle caratteristiche per ogni operazione. Una mappa di caratteristiche è una matrice di valori che rappresenta l’informazione presente in quella porzione di immagine. Le mappe create vengono passate ai livelli di pooling che riducono le dimensioni delle mappe di caratteristiche ed il numero di parametri della rete. I dati così organizzati non si prestano all’elaborazione in un livello completamente connesso, motivo per cui le matrici vengono trasformate in un vettore monodimensionale prima di essere passate al livello completamente connesso. Nel caso del PAD il livello finale sarà di classficazione binaria, cioè con sole due opzioni di classificazione (Live o Fake). L’innovazione maggiore portata da questi modelli è la capacità di inferire automaticamente dal dataset le caratteristiche delle immagini, oltre al minore tempo di calcolo necessario dopo l’addestramento della rete.



#### Figura 2.3: Struttura di una rete convoluzionale.

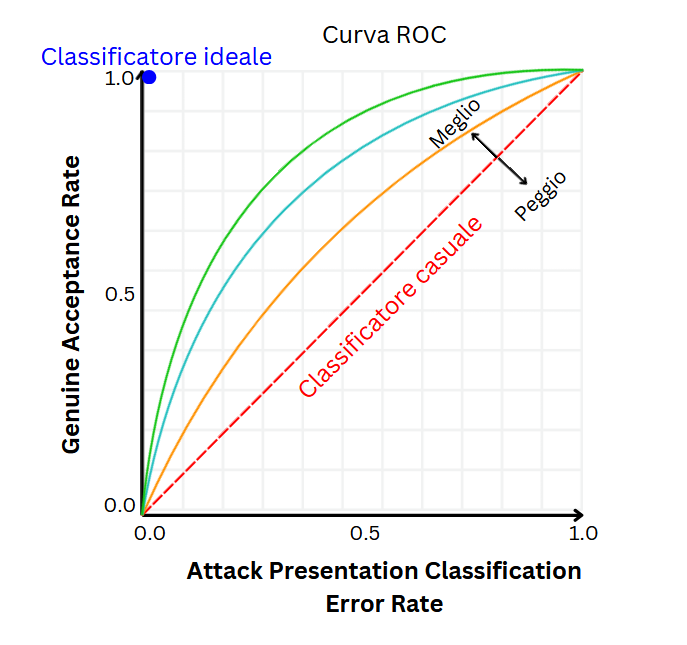
## 2.4 Metodi di valutazione

I sistemi PAD sono soggetti a vari tipi di errore che possono rendere il sistema fallace o di difficile usabilità e sono valutati secondo varie metriche. Oltre alle qualità delle immagini citata precedentemente, il sistema deve essere efficiente, efficace ed user-friendly (facile da usare, affidabile, ecc.), per avere possibilità di utilizzo nel mondo reale [13].

Le metriche sulle performance dei PAD valutano la quantità di un tipo di errore che viene commesso dal sistema. Le metriche principali definite da [22] sono:

* APCER Attack Presentation Classification Error Rate – Percentuale di Presentation Attack in cui l’attaccante viene classificato come utente genuino, generalmente noto come FAR (False Acceptance Rate), in cui l’etichetta positiva è attribuita agli utenti genuini.
* BPCER Bona fide Presentation Classification Error Rate – Percentuale di tentativi di accesso genuini classificati come Presentation Attack, generalmente noto come FRR (False Rejection Rate)

La variazione dei parametri del modello porta a scegliere un compromesso tra APCER e BPCER. Se si sceglie di focalizzarsi sulla sicurezza, solo in caso di alta correlazione tra l’immagine in input ed un’impronta nel dataset sarà consentito l’accesso, con valori alti di BPCER e inficiando sull’usabilità del sistema. Nel caso fosse necessaria una maggiore usabilità i parametri possono essere variati per consentire l’accesso per campioni con una minore correlazione con template del dataset, con un alto valore di APCER e inficiando sulla sicurezza. La bontà di un modello di Machine Learning può essere visualizzata tramite il grafico della curva ROC (Figura 2.4). La curva ROC nel caso dei PAD è formata come il rapporto tra APCER e GAR (Genuine Acceptance Rate), calcolato come 1 – BPCER. L’integrale dell’area sottesa dalla curva ROC prende il nome di AUC (Area Under the Curve), che viene utilizzato come ulteriore metro di confronto tra modelli. Un modello perfetto otterrà un punteggio AUC di 1, mentre un modello che classifica casualmente tenderà ad un valore di AUC di 0,5.



#### Figure 2.4: Curva ROC, in cui è rappresentato il caso in cui un classificatore classifichi in maniera casuale e, come il lancio di una moneta, avrà una precisione del 50%.

## 2.5 Dataset

Un dataset è una collezione strutturata di dati, spesso organizzati in forma tabellare, in cui ad ogni riga corrisponde un elemento del dataset e ad ogni colonna corrisponde un attributo degli elementi. Gli attributi in un dataset possono essere numerici, categorici, ordinali oppure testuali e ne descrivono gli elementi. Nel caso delle immagini, ogni immagine è un elemento del dataset e a questa sono assegnati i suoi attributi. Ad esempio, i dataset per i sistemi PAD sono dataset di immagini che possono presentare diversi attributi, come sensore di acquisizione, materiale o altro, da cui l’algoritmo impara delle informazioni. La comparazione di impronte è un processo in cui l’algoritmo deve predire il tipo di immagine che viene fornita in input, perciò i campioni dati in addestramento avranno sicuramente un attributo che definisce se siano vere o false, detto etichetta o label, che ne definisce la classe. Questi processi sono detti compiti di classificazione. L’etichetta deve essere apposta manualmente, preferibilmente dallo stesso individuo che effettua la raccolta dei dati. Se degli attributi non sono presenti per alcuni campioni vi è un problema di completezza che può influire sull’addestramento e sui risultati.

Una delle principali caratteristiche di un dataset è il suo bilanciamento, ovvero quanto equamente è rappresentata ogni classe. Un dataset sbilanciato porta ad una bassa capacità di generalizzazione [35] e ad un aumento degli errori di classificazione, ovvero campioni per cui l’algoritmo di Machine Learning predice una classe sbagliata. Il bilanciamento è uno dei problemi principali per i dataset dei PAD, poiché per creare un dataset di grandi dimensioni bisogna ottenere sia un grande numero di impronte sia delle loro falsificazioni in diversi materiali. Un’ulteriore problema specifico dei dataset per i PAD è l’esaustività, ovvero la presenza di varie impressioni dello stesso dito, che permette una migliore capacità di generalizzazione, e della corrispondente impronta falsificata.

L’ottenimento delle falsificazioni richiede un grande dispendio di tempo ed energia, visto che, idealmente, per ogni impronta del dataset si dovrebbe avere la sua controparte nel maggior numero di materiali possibili. Si può notare nella Tabella 2.1 che questo non è possibile neanche nei dataset più utilizzati nel campo. La creazione di un numero esteso di falsificazioni necessita di esperienza e strumentazione, inoltre il numero di materiali rappresenta una problematica in sé, visto il numero teoricamente illimitato con cui si possono riprodurre le impronte.

A causa della sensibilità delle biometrie non è disponibile pubblicamente un gran numero di dataset di impronte digitali, questa è un’importante problematica soprattutto per i modelli Deep Learning. Il numero di dataset di impronte ha subito un ulteriore diminuzione in seguito alla creazione del GDPR, in cui viene richiesta l’autorizzazione esplicita da parte dell’individuo per il trattamento dei suoi dati e ulteriori protezioni per i dati sensibili, di cui le biometrie fanno parte.

Nella Tabella 2.1 si può notare come anche i dataset più utilizzati nel campo abbiano un numero molto ristretto di campioni.

#### Tabella 2.1. Caratteristiche di alcuni dataset molto utilizzati per l’addestramento di PAD e GestSpoof, un dataset per gli approcci dinamici di Spoof Detection.

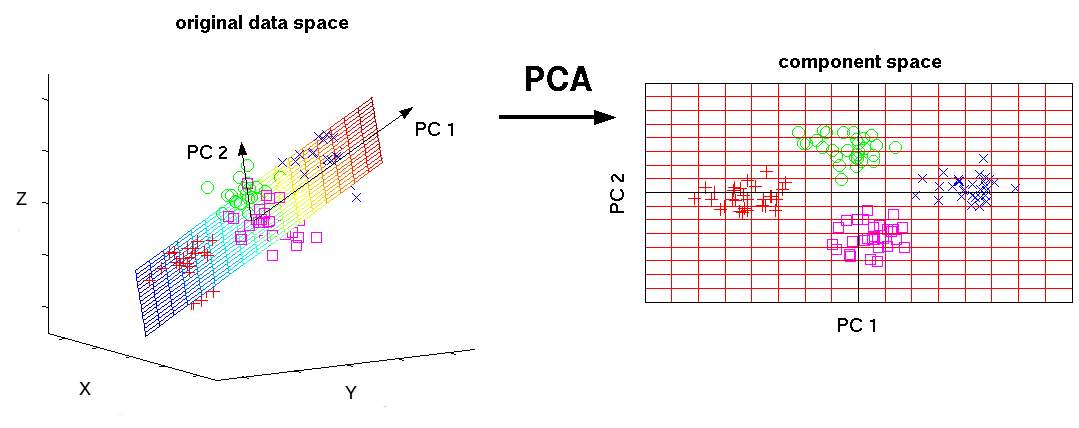
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nome Dataset** | **Tipo di Sensori** | **Materiali finti** | **Campioni totali** | **Campioni veri** | **Campioni finti** |
| LivDet 2017  [25] | Ottico, termico | Lattice, Gelatina, Ecoflex, Ecoflex liquido, Gomma siliconica, Colla per legno | 18000 | 8100 | 9900 |
| LivDet 2023  [23] | Ottico, ibrido | Lattice,  Silicone, Colla per legno, Gomma | 9750 | 3750 | 6000 |
| MSU FPAD  [24] | Ottico | Ecoflex,  Plastilina,  Stampa su carta nera 2-D,  Pellicola trasparente | 19500 | 9000 | 10500 |
| GestSpoof  [26] | Video | Gomma siliconica, EcoFlex, Gelatin | 3680 | 920 | 2760 |

## 2.6 Limiti degli attuali sistemi PAD

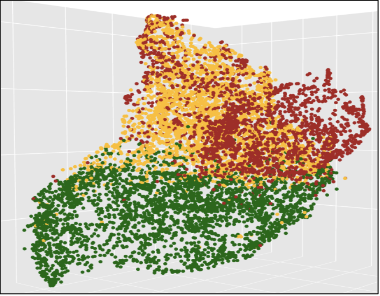
Nonostante i sistemi PAD siano diventanti più performanti e responsivi nel corso degli anni, soffrono ancora di alcune problematiche limitanti.

Gli attacchi di presentazione, come detto precedentemente, richiedono conoscenze tecniche per poter replicare efficacemente un’impronta, ma le possibili variazioni della tecnica rendono il numero di attaccanti e di materiali utilizzabili illimitato. Da questo deriva la necessità di un’altissima capacità di generalizzazione del sistema, difficilmente ottenibile oggi a causa della mancanza di dataset abbastanza grandi e rappresentativi. I sistemi PAD inoltre devono essere rapidi ed integrati con una minima riduzione nei tempi di risposta nel riconoscimento, perciò risultano limitati nella complessità. Un ulteriore problema è rappresentato dalla costante evoluzione degli attacchi, per cui un nuovo materiale ipotetico potrebbe essere non riconosciuto correttamente dal PAD.

Una possibile soluzione alle problematiche sopra citate possono essere gli algoritmi di Deep Learning [88]. Gli algoritmi di Deep Learning generano vettori di feature ad alta dimensionalità, rendendole effettivamente illeggibili per un utente umano e rendendo difficile il tracciamento degli attributi. Con tecniche per la riduzione delle dimensionalità dei dati, come la Principal Component Analysis (PCA), è possibile ridurre la dimensionalità dei vettori di feature mantenendo la loro relazione [18] (Figura 2.5). Questo risulta un utile strumento per visualizzare come vengono mappati i diversi campioni nello spazio ridotto e, osservando i cluster (gruppi) che si formano, è possibile vedere come questi raccolgano campioni di impronte false in base al loro materiale (Figura 2.6). Questo è dovuto al fatto che materiali diversi hanno caratteristiche diverse, come la porosità del materiale o il modo in cui sporcano il sensore, e quindi feature differenti. Intuitivamente si può pensare che ogni tipo di materiale sia mappabile in questo spazio e, perciò, che in questo spazio si possa trovare qualsiasi materiale. I punti nei quali non vengono mappati campioni saranno quelli per cui il materiale ha caratteristiche diverse rispetto a quelli visti in addestramento, ma un sistema di Deep Learning che non ha mai visto questo materiale potrebbe categorizzarlo come non finto, visto che è comunque differente dai materiali conosciuti dal modello. Tramite i metodi di Data Augmentation e di generazione sintetica basati sul Deep Learning potrebbe essere possibile creare campioni dei dati con variazioni nelle caratteristiche che andrebbero a popolare gli spazi vuoti, migliorando le capacità del sistema di riconoscimento.



#### Figura 2.5 Esempio di riduzione dimensionale con la PCA, in cui dati in uno spazio a tre dimensioni vengono mappati in uno spazio bidimensionale, mantenendo la relazione tra loro.



#### Figura 2.6 Grafico t-SNE per la visualizzazione 3D di dati in molte dimensioni. Mostra come vengono mappati campioni live (verdi) e campioni falsi conosciuti (rossi) e non conosciuti (gialli).

I metodi di Data Augmentation possono essere impiegati per la risoluzione dell’esaustività del dataset [87], fornendo la possibilità di creare nuove visualizzazioni delle dita presenti nel dataset, sia per le impronte reali che falsificate.

I metodi di generazione sintetica possono anch’essi essere una soluzione alle problematiche del dataset, creando nuove impressioni delle impronte già presenti oppure creando nuove impronte non presenti nel dataset, ampliandolo. Le reti generative possono anche essere utilizzate come tecnica di Data Augmentation per trasferire lo stile di un’immagine in un'altra (Figura 2.7): questa tecnica viene chiamata Neural Style Transfer [41].

Tramite lo Style Transfer è possibile apporre lo stile di un materiale su un’impronta reale, ottenendo un’impronta falsificata sintetica. La combinazione di generazione sintetica e Data Augmentation potrebbe risolvere le problematiche legate a dataset ristretti, difficoltà nell’ottenimento delle impronte e delle loro falsificazioni, fornendo un dataset completo, rappresentativo ed esaustivo per ogni impronta, reale o meno.

Immagine che contiene testo, schermata, grafica, arte

Descrizione generata automaticamente

#### Figura 2.7 Trasferimento dello stile da un quadro ad una foto. L’immagine risultante sarà la foto rappresentata con lo stile del quadro.

# Capitolo 3

# Data Augmentation

In questa tesi è stata citata la necessità di una grande mole di dati per gli algoritmi di Machine Learning, e tanto più per i modelli di Deep Learning, per avere un addestramento ottimale del modello. L’attuale mancanza di dataset di grandi dimensioni esaustivi porta a modelli con scarse capacità di generalizzazione, mentre si è notato come un dataset esteso e rappresentativo porti anche i modelli più semplici ad avere migliori performance [35]. Per ovviare a queste problematiche si possono utilizzare varie tecniche per la manipolazione dei dati che vengono chiamati metodi di Data Augmentation. Lo scopo di questi processi è di aumentare la quantità, la diversità e la qualità dei dati senza cambiare la distribuzione dei dati o aggiungere nuove caratteristiche.

## 3.1 Tecniche di Data Augmentation

Nonostante il grande progresso compiuto negli ultimi anni nel campo dell’intelligenza artificiale, il dataset con cui si addestrano i modelli resta il fattore più importante. La difficoltà nell’ottenere grandi quantità di dati ha portato allo sviluppo di tecniche per migliorare artificialmente i dati a disposizione, chiamate tecniche di Data Augmentation [87]. La Data Augmentation ha lo scopo di aumentare la quantità e la varietà dei dati di addestramento senza dover raccogliere nuove immagini o informazioni; questo può avvenire tramite vari tipi di operazioni che variano i valori e/o nella struttura dei dati da analizzare. L’utilizzo di queste tecniche rende il modello più robusto, cioè più resistente alle variazioni ambientali e alle differenze tra campioni, migliorando le capacità di generalizzazione. Con l’aumento della capacità di generalizzazione si riduce il rischio che il modello impari la distribuzione dei dati in modo troppo dettagliato: questo problema è chiamato overfitting ed è tipico di reti neurali di dimensioni troppo ridotte o che sono state addestrate su dataset non sufficientemente estesi. Nel caso di reti di dimensioni ridotte il rischio di overfitting può essere ridotto con varie tecniche [36,38], ma queste non possono risolvere i problemi legati alle caratteristiche del dataset. La Data Augmentation risulta utile per questo scopo a prescindere dalla profondità della rete, ma le tecniche applicabili per una rete dipendono dalla tipologia di dato che questa analizza; in questa tesi verranno trattati soltanto i metodi riguardanti le manipolazioni di immagini.

## 3.2 Data Augmentation nelle immagini

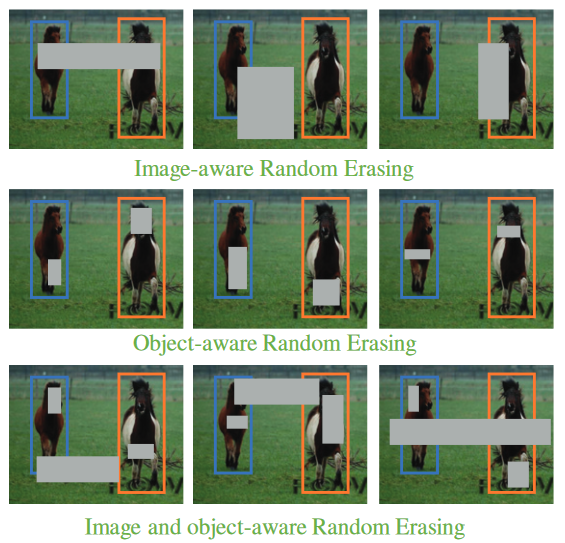
Le rilevazioni ottenute dai sensori di rilevamento di impronte digitali sono utilizzate sotto forma di immagini sia nei sistemi di riconoscimento, sia negli algoritmi classici e nelle reti convoluzionali su cui si basano i PAD. Le immagini sono composte da pixel, unità numeriche che contengono un valore rappresentante il colore, contenute in una griglia. Le immagini sono quindi un tipo di dato ben strutturato su cui possono venir applicate manipolazioni del valore dei pixel e della loro posizione.

I metodi di Data Augmentation per le immagini permettono di creare variazioni di queste tramite diverse tecniche basate sulla singola immagine, su multiple immagini o sull’intera popolazione del dataset.

### 3.2.1 Augmentation di immagini singole sul valore

1. Cancellazione dei pixel

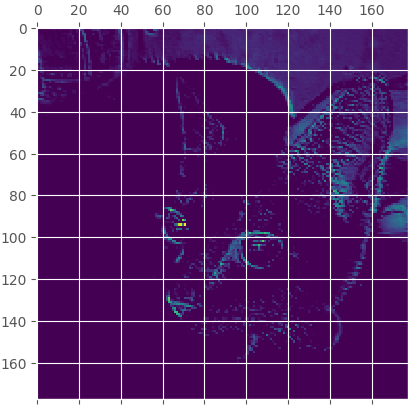
Tecnica basata sulla semplice idea di nascondere parte dell’immagine in input o dello spazio delle caratteristiche [45]. Generalmente implementata con la selezione di un rettangolo di pixel che viene cancellato e poi riempito con un dato colore o con del rumore Gaussiano. Se l’area da cancellare è scelta casualmente, la tecnica è chiamata Image-aware Random Erasing (IRE) [46] (Figura 3.1); se viene cancellato un oggetto individuato da un riquadro di delimitazione, la tecnica è chiamata Object-aware Random Erasing (ORE) [46]. Un’ ulteriore versione di questa tecnica, chiamata Region-aware Random Erasing (RRE) [47], implementa un ORE con dimensioni limitate superiormente per poi cancellare casualmente le zone dello sfondo. Tutte le versioni viste finora agiscono nel dominio dei dati, mentre Random Erasing Frequency (REF) [67] agisce sulle frequenze, usando la trasformata di Fourier per trasformare l’immagine, applica il Random Erasing e utilizza la discreta inversa della trasformata per restituire l’immagine. Questi metodi di Random Erasing appongono una o due grandi maschere rettangolari all’immagine, ma esistono versioni più flessibili come la Hide-And-Seek [68], che rimuove varie parti dell’immagine per forzare il modello ad imparare da parti meno discriminanti dell’input. La versione Grid-Mask [66] crea una griglia di maschere quadrate da apporre all’immagine e può raggiungere un ottimo equilibrio tra cancellazione e mantenimento dei dati.



#### Figura 3.1 Un esempio di IRE (alto), ORE (centro) e la combinazione delle due tecniche (basso). Gli oggetti rionosciuto sono circondati dal riquadro di delimitazione.

1. Trasformazione fotometrica

A differenza delle tecniche viste, Photometric Transformation non cancella i pixel ma ne varia il valore, per esempio variando la luminosità, sostituendo il colore con quello opposto (Color Inversion), modificando i valori dei canali di colore (Color Casting), aggiungendo rumore (Noise Injection) [90]. Con queste tecniche è possibile creare una grande quantità di dati sintetici in grado di aumentare la robustezza del modello, senza tener conto delle relazioni tra pixel. Metodi che tengono conto dei pixel circostanti sono l’equalizzazione dell’istogramma, che varia il contrasto, e il bilanciamento dei bianchi, che varia l’illuminazione dell’immagine. Diversi tipi di filtri possono essere applicate ad un’immagine per crearne una nuova, come i filtri Kernel, che generano convoluzioni dell’immagine originale (Figura 3.2). Si può trasformare l’immagine in scala di grigi (Greyscale) per far concentrare il modello sulle forme presenti più che sui colori e per diminuire la dimensionalità delle immagini, alleggerendo il modello. Può anche essere applicata la tecnica inversa al Greyscaling, il Greyscale-to-Color Conversion (GCC), utile nel caso ci siano delle relazioni tra i canali di colore o per quei modelli che sono stati addestrati su immagini a colori.



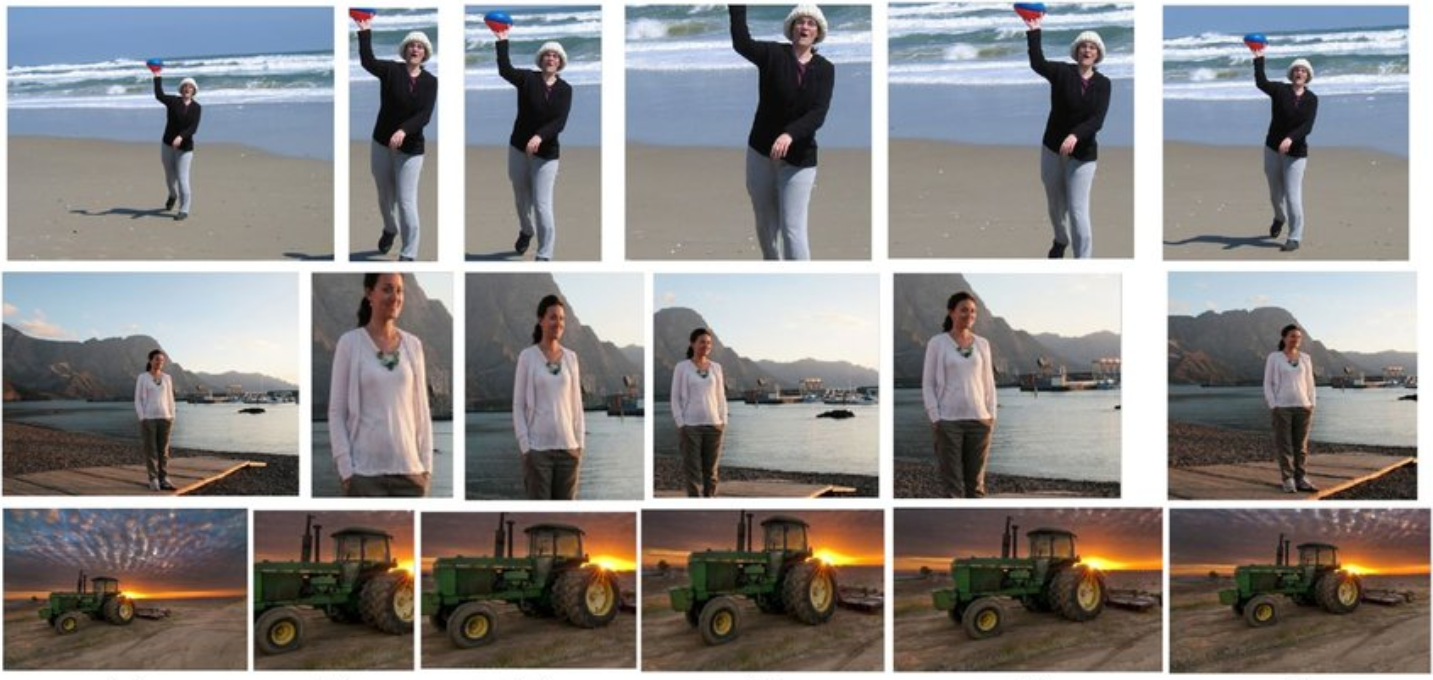
#### Figura 3.2 Convoluzione di immagini. Una delle 32 immagini risultanti (a destra) da una convoluzione di un'immagine a colori di dimensioni 180x180 pixel con un filtro Kernel di dimensioni 5x5.

### 3.2.2 Augmentation di immagini singole sulla struttura

1. Ritaglio dell’immagine

Queste tecniche si basano sulla selezione di un’area dell’immagine, rimuovendo il resto. Alcune di queste tecniche sono utilizzate per ridurre la dimensionalità dei dati e snellire il modello, ma sono utilizzabili anche per la Data Augmentation visto che il ritaglio può generare diverse immagini, selezionando diverse aree della stessa immagine e rimuovendo parti meno utili dello sfondo (Figura 3.3). I metodi più semplici utilizzano una selezione manuale oppure casuale dell’area, mentre quelli più avanzati sono basati sull’Attention [42], tecnica che attribuisce un punteggio di attenzione a diversi settori dell’immagine per individuare i più significativi. I settori possono essere individuati da metodi automatici che utilizzano l’entropia, il rateo dell’area e la distanza dal centro dell’immagine per suddividere e poi classificare i settori, ma tralasciano il problema della ricerca del rettangolo di ritaglio ottimale. Un’alternativa a questi metodi sono quelli interattivi che, utilizzando un sensore di monitoraggio oculare, individuano i punti con maggior focus del soggetto per inferire le zone di maggior interesse. Questa metodologia è un compromesso tra l’implementazione manuale e quella completamente automatica.

Nei metodi basati sull’estetica, [64] suddivide l’immagine in regioni soggetto e regioni sfondo, da cui estrae le caratteristiche per poi stimare e combinare le probabilità a posteriori e attribuire un valore di qualità. È possibile integrare questi metodi con algoritmi di Reinforcement Learning [65]: queste reti profonde utilizzano un agente che svolge azioni in un ambiente e a cui viene fornito un punteggio in base all’output che ne modifica il prossimo ciclo di azioni, fino a trovare quello ottimale. Nei metodi basati sull’estetica il Reinforcement Learning è usato per risolvere il problema del miglior ritaglio. Un ulteriore metodo [91] utilizza un approccio basato sul cambiamento apportato all’immagine dall’operazione di ritaglio per valutarne le prestazioni su specifiche caratteristiche.



#### Figura 3.3 Creazione di variazioni di immagini tramite differenti tecniche di ritaglio.

1. Trasformazioni geometriche dell’immagine

Sono i metodi più classici di Data Augmentation, che variano le relazioni spaziali tra pixel senza variarne il valore. Questi metodi creano nuove immagini per il training tramite rotazioni, traslazioni, zoom o specchiamento (Figura 3.4) (trasformazioni affini che non cambiano il rapporto tra i pixel e riflettono il mondo reale) oppure trasformazioni non affini come il cambio di prospettiva (che cambiano il rapporto spaziale tra i pixel), introducendo un ulteriore varianza nel dataset. Possono essere introdotte distorsioni e stretching di porzioni dell’immagine che aiutano ulteriormente il modello durante l’apprendimento, espandendo lo spazio dell’input e consentendo una maggiore capacità di generalizzazione.



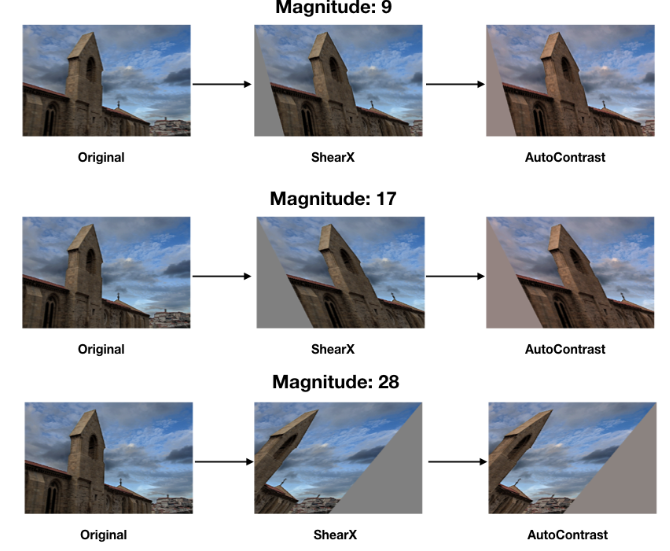
#### Figura 3.4 Trasformazioni geometriche. Immagini create a partire dall'immagine originale tramite rotazioni, zoom e specchiamento.

### 3.2.3 Augmentation di immagini singole su valore e struttura

Questo tipo di tecniche di Data Augmentation mirano a migliorare il processo tramite l’utilizzo del Reinforcement Learning per trovare le politiche, cioè la sequenza di azioni, migliori. Gli algoritmi in questa tipologia sono chiamati algoritmi di Data Augmentation automatica.

L’algoritmo AutoAugment [55] è uno dei metodi più significativi di questa tipologia: rappresenta la Data Augmentation come un processo decisionale per ricercare le scelte migliori. Un’operazione può essere qualsiasi tecnica basata sul valore o basata sulla struttura. Una sotto-politica consiste in due operazioni con le loro probabilità di applicazione e magnitudine con cui queste vengono applicate. I loro valori formano lo spazio di ricerca, in cui un agente sceglie le sotto-politiche per formare una politica. Una politica è composta da più sotto-politiche. Le immagini sono processate in base alla politica scelta, creando varie immagini che vengono fornite alla rete.

AutoAugment misura le perfomance della rete e le utilizza per variare l’agente, che viene così addestrato a selezionare le trasformazioni ed i parametri migliori, che verranno utilizzati su tutte le immagini e potranno essere anche applicati su altri dataset. Esistono variazioni di questo algoritmo come Fast AutoAugment [54], che snellisce il processo di ricerca delle sotto-politiche, permettendone un numero maggiore; il chè dovrebbe permettere una maggiore capacità di generalizzazione. RandAugment [56] rimuove la necessità di uno spazio di ricerca, scegliendo casualmente le operazioni, allegerendo la memoria e diminuendo il tempo richiesto (Figura 3.5). Population Based Augmentation (PBA) [57] utilizza una pianificazione delle trasformazioni anziché una politica per migliorarne l’efficacia.



#### Figura 3.5 Un esempio di immagine augmentata con RandAugment, con sotto-politiche applicate con magnitudini differenti.

### 3.2.4 Augmentation di immagini multiple sul valore

Questa tipologia di augmentation su immagini multiple compie una manipolazione aritmetica (ad esempio interpolazione) dei colori provenienti da immagini diverse.

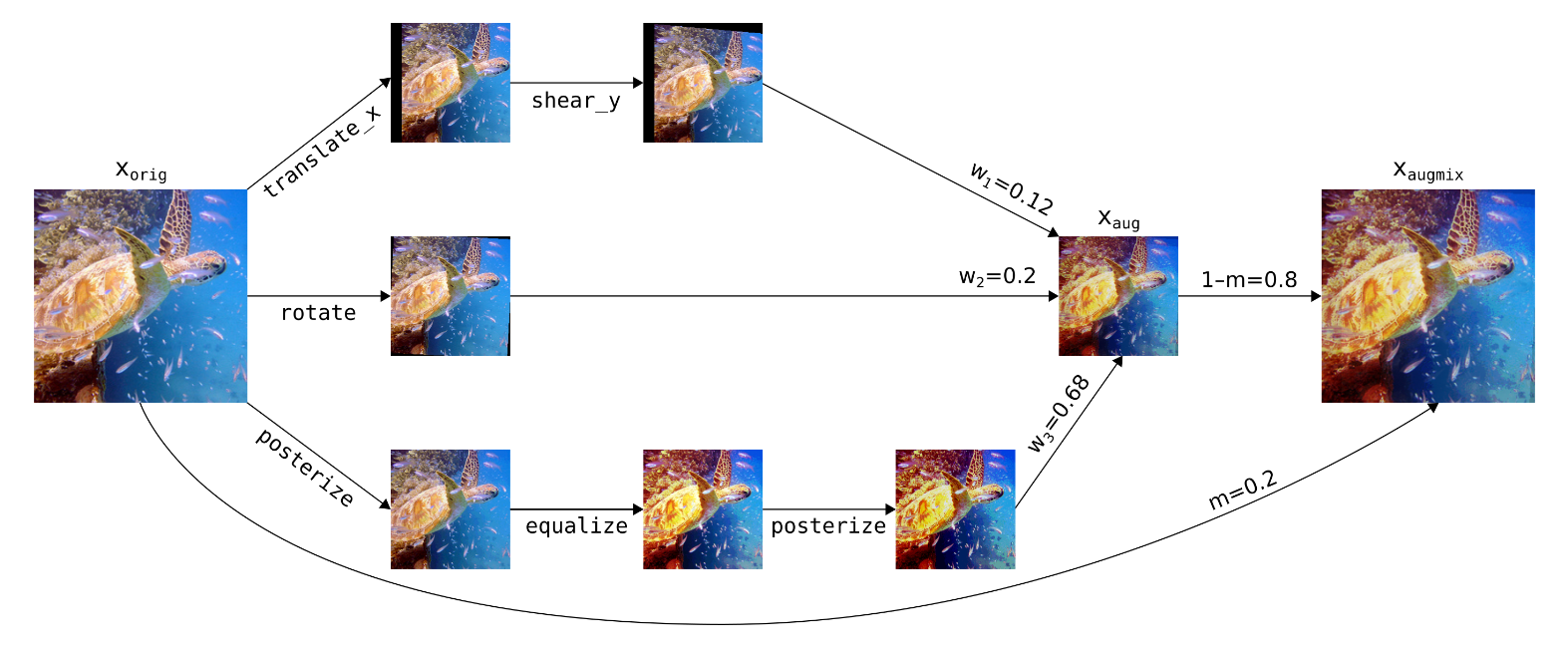
1. Mixup di immagini

Questa categoria di tecniche combina i pixel di due immagini. Ogni pixel risultante è prodotto dall’interpolazione dei pixel delle immagini originali. Mixup [59] mescola due immagini casuali in input, i pixel di ogni immagine contribuiscono ad una parte della nuova immagine secondo la formula

˜x = 𝜆x𝑖 + (1 − 𝜆)x𝑗 , ˜y = 𝜆y𝑖 + (1 − 𝜆)y𝑗

dove (xi,yi) e (xj,yj) sono le coordinate di due immagini etichettate del dataset e 𝜆 è un valore compreso tra 0 e 1.

AugMix [63] invece applica tre operazioni di Augmentation ad un’immagine in input e combina le tre immagini così create per fornire un nuovo campione (Figura 3.6). In entrambi i metodi il fattore di mescolamento è uguale in tutta l’immagine, restando consistente per tutte le regioni. Negli approcci di miscelamento per regione il fattore di mescolamento cambia per le diverse regioni. Per esempio SmoothMix [58] applica una soft-edged mask, ovvero una maschera in cui i valori ai bordi vanno a diminuire in base ai parametri della maschera. Due metodi più sofisticati sono Co-mixup [60] e Puzzle mix [61], che utilizzano entrambi la salienza per decidere la magnitudine del mescolamento di diverse regioni, potenzialmente migliorando le performance.



#### Figura 3.6 Il processo di creazione di campioni di AugMix. AugMix crea tre variazioni dell’immagine originale con operazioni campionate casualmente, per poi mescolare le tre risultanti in un’immagine finale.

La maggior parte dei metodi di mescolamento creano nuove immagini con “etichette morbide” che, a differenza delle normali “etichette dure”, non sono valori interi ma un punteggio di probabilità di appartenenza ad una classe. Sample Pairing utilizza l’etichetta di una delle immagini originali, senza utilizzare l’altra. Remix[62] è stato creato per risolvere il problema dello sbilanciamento delle classi assegnando l’etichetta della classe minoritaria alle nuove immagini.

1. Neural blending

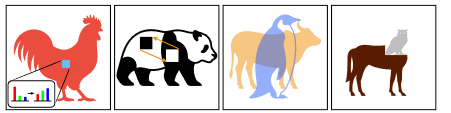
Anziché mescolare le immagini con metodi aritmetici, questi metodi utilizzano le reti neurali. AutoMix integra un Mix Block [43], un modulo basato sulla cross-Attention per selezionare i pixel da mescolare, nel training, creando un ponte tra la selezione della politica di mescolamento e l’ottimizzazione del modello. Il modulo di Augmentation è ottimizzato insieme alla mappa di classificazione delle caratteristiche. La funzione ed il rateo di mescolamento saranno aggiornati in base alla funzione di perdita della classificazione. La tecnica Smart Augmentation prende due immagini dall’input e utilizza una sotto-rete convoluzionale per restituire un output della stessa dimensione dell’input, ovvero da due immagini ne viene prodotta una nuova.

### 3.2.5 Augmentation di immagini multiple basate sulla struttura

Metodi di questo tipo combinano immagini in base alla posizione dei pixel e alla sottostruttura dell’immagine. Prendono il nome di Image Patching, in riferimento al processo con cui questi metodi mettono insieme piccoli frammenti di molte immagini oppure aggiungono una parte di un’immagine ad una seconda immagine.

1. Image Patching

RICAP (Random Image Cropping and Patching) [44] ritaglia 4 immagini casualmente e le unisce per creare una nuova immagine. CutMix [48], basata su Cutout [45], utilizza solo due immagini, ma le regioni ritagliate sono sostituite da regioni ritagliate da altre immagini. Attentive CutMix [49] migliora il metodo con l’aggiunta di un sistema di estrazione delle caratteristiche che seleziona zone importanti o significative delle immagini e incolla queste porzioni in altre immagini. Le tecniche di Image Patching non sono limitate a ritagli rettangolari, metodi come CowMask [50] possono utilizzare anche ritagli con forma irregolare. Nella Figura 3.6 si possono vedere alcune delle operazioni possibili con questi metodi.



#### Figura 3.6 Una dimostrazione delle diverse manipolazioni possibili con i metodi di Image Patching. In ordine da sinistra a destra: Trasformazione di immagine individuale tramite valore, trasformazione di immagine individuale tramite struttura, mescolamento di immagini sul valore, combinazione di immagine tramite struttura.

### 3.2.6 Augmentation di immagini basate sulla popolazione

1. Generazione di immagini standard

Le reti GAN (Generative Adversarial Network) [51] sono un tipo di reti neurali profonde che permettono di creare nuove immagini. La generazione parte da una distribuzione di rumore ed un dataset iniziale per l’addestramento, che viene utilizzato per giudicare la verosimilità delle immagini generate. Le GAN sono ormai ampiamente utilizzate per la Data Augmentation, soprattutto in ambienti in cui la mole di dati disponibili è scarsa. Oltre a creare nuovi campioni, le GAN sono utilizzate per rimuovere il rumore dallo sfondo e per migliorare il dataset [89]. Inoltre possono generare differenti punti di vista di un’immagine, aggiungendo varianza al dataset.

1. Generazione di immagini esogena

Il Neural Style Transfer [41] fa parte di questa categoria, la tecnica si basa sull’apprendere lo stile artistico da un gruppo di dati di immagini e utilizzarlo per rappresentare altre immagini. Neural Style Transfer può essere usato per imitare lo stile di un particolare artista, aiutando il modello a imparare una rappresentazione più generale dell’argomento. Un altro utilizzo della tecnica può essere importare il clima o l’illuminazione da immagini di riferimento nel dataset target; questo è particolarmente utile quando è difficile raccogliere sufficienti immagini in un certo ambiente.

Un’altra tecnica possibile è il modellamento grafico, che consiste nella creazione artificiale tramite software, come Blender o Unity, di modelli dei dati e può essere utilizzato per produrre dataset adatti al training di modelli di intelligenza artificiale. Questi dataset hanno vari vantaggi su quelli tradizionali, per esempio non c’è bisogno di un lavoro di raccolta dati, e, con l’applicazione di diverse pose, aspetti e condizioni di luminosità, il modellamento di immagini può generare un’altissima diversità tra i campioni. Nonostante la creazione manuale di un modello dei dati richieda una conoscenza del modello e del software, un ulteriore vantaggio è che l’etichettatura dei campioni sarà sempre corretta.

## 3.3 Applicazioni su FPAD

Tra i metodi di Data Augmentation più semplici applicate ai PAD spiccano le trasformazioni geometriche, le trasformazioni fotometriche e la noise injection. Queste sono facilmente implementabili, forniscono una buona varietà di immagini, hanno basso costo e basso tempo di computazione, per cui vengono ampiamente utilizzate [92]. Grazie alle trasformazioni geometriche, come rotazioni e ridimensionamenti, si ottiene una maggiore variabilità delle impronte simulando le differenze tra rilevazioni reali. Le trasformazioni fotometriche permettono di simulare diverse condizioni climatiche di rilevazione, come umidità, pressione e luminosità. L’aggiunta di rumore tramite la noise injection ha la doppia utilità di aggiungere variabilità alle immagini e rendere il modello più resistente al rumore [90]. Queste tecniche, tuttavia, non forniscono un numero di campioni ed una variabilità sufficienti per risolvere il problema dei dataset ristretti. Le tecniche di mescolamento di immagini, neural blending e di patching portano alla perdita dell’identità e della struttura dell’impronta, rendendo problematica l’identificazione e potrebbero portare ad impressioni non realistiche, motivo per cui risultano non applicabili al dominio dei dati. Tecniche più avanzate come l’utilizzo di GAN [20,53] e style transfer [41] possono fornire grandi vantaggi se adattate al dominio dei dati, queste tecniche sono spiegate nel dettaglio nel Capitolo 4 data la loro capacità di generazione sintetica.

## 3.4 Limiti della Data Augmentation

I metodi che sono stati discussi in questo Capitolo permettono un grande aumento del numero di campioni veri e falsi, della loro variabilità e qualità. Questi metodi, però, non sono ancora una soluzione a tutte le problematiche di cui soffrono oggi i sistemi PAD. Le immagini create con le trasformazioni geometriche sono teoricamente infinite, ad esempio ruotando il campione originale di una frazione di grado per volta si creerebbe un enorme numero di campioni, ma buona parte di questi fornirebbero solo informazioni ridondanti. Le proprietà delle impronte generate sono strettamente legate alla distribuzione iniziale dei dati, inoltre con queste tecniche non è possibile affrontare la problematica dei materiali sconosciuti. I sistemi che utilizzano solo tecniche di Data Augmentation non possono essere definiti sufficientemente resistenti ad un attacco di presentazione, principalmente a causa della difficoltà nel rilevare materiali e sensori differenti da quelli presenti nel dataset.

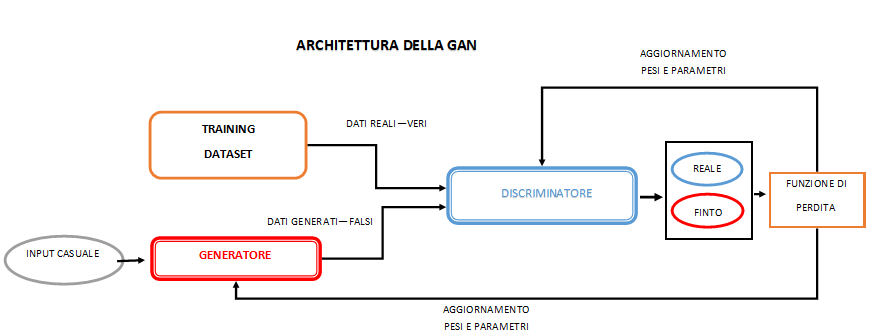
Una possibile soluzione alle problematiche dei materiali, dei sensori e del numero di campioni si può trovare nell’utilizzo congiunto delle tecniche di Data Augmentation con i modelli di generazione sintetica di impronte che verranno discusse nel successivo Capitolo.

# Capitolo 4

# Generazione sintetica per l’addestramento di rilevatori di attacchi di presentazione

Le tecniche di Data Augmentation viste nel Capitolo 3 forniscono vari strumenti per migliorare i dataset, ma soffrono di importanti limitazioni. Un’alternativa alla Data Augmentation per ovviare ai difetti di un dataset è la generazione sintetica di dati. Con generazione sintetica si intende una tecnica che permette di produrre nuovi campioni di dati con algoritmi e tecnologie di Intelligenza Artificiale generativa. I dati possono essere usati per ampliare un dataset in modo che, a differenza della Data Augmentation, siano introdotte nuove caratteristiche, variando la distribuzione dei dati originale. Le moderne tecniche di intelligenza artificiale, soprattutto le reti neurali GAN [51] e DDPM (Denoising Diffusion Probababilistic Models) [69] consentono di creare un grande numero di campioni con una grande varietà, sia di impressioni che di materiali. Il loro impiego, inoltre, risolve il problema della protezione della privacy dei possessori dei dati e quindi permettono l’utilizzo di dati sensibili come le biometrie. Per ottenere delle impressioni realistiche è, però, necessario un addestramento stabile e computazionalmente costoso, al fine di evitare allucinazioni o artefatti.

## 4.1 Modelli GAN per la generazione sintetica

Le reti GAN (Figura 4.1) sono reti neurali di tipo Deep in grado di imparare distribuzioni di dati complesse a partire da un dataset e da una distribuzione di rumore [51]. Sono formate da due reti con compiti distinti: il generatore ed il discriminatore. Il generatore, partendo da rumore, genera delle immagini che vengono inviate insieme ad immagini reali al discriminatore; questo ha il compito di distinguere le immagini vere da quelle generate. Il discriminatore viene addestrato su un dataset, in questo caso di impronte digitali, mentre il generatore crea delle immagini a partire da una distribuzione di rumore. Le immagini create al primo ciclo di generazione saranno molto lontane da quelle vere, motivo per cui il discriminatore non avrà difficoltà a distinguerle. L’output della rete discriminatrice sarà utilizzato per variare i pesi del generatore in modo che, nelle prossime esecuzioni, generi delle immagini sempre più simili a quelle del dataset, fino ad eludere il classificatore. La funzione di perdita di queste reti è detta funzione di perdita avversaria perché, dato che il generatore produrrà dati sempre più vicini alla distribuzione del dataset, la sua funzione di perdita partirà da un valore alto e diminuirà gradualmente. Il discriminatore inizialmente classificherà tutte le immagini correttamente, ma poi commetterà sempre più errori, quindi il suo valore di perdita sarà inizialmente molto basso, per poi aumentare costantemente. La funzione di perdita avversaria è quindi il valore di perdita del discriminatore, che il sistema cerca di massimizzare, sommato ad 1 meno il valore di perdita del generatore, che il sistema cerca di minimizzare. Generalmente le classiche reti GAN non consentono la variazione di attributi specifici e sono soggette ad allucinazioni, per cui non è possibile produrre impressioni molto realistiche o gestire il processo di creazione. Modelli moderni più avanzati e performanti per la creazione di record sintetici, come SpoofGAN [53] e FPGAN-Control [20], riescono ad attenuare queste problematiche e produrre immagini molto realistiche. 

#### Figura 4.1 Architettura di un Generative Adversarial Network (GAN).

### 4.1.1 SpoofGAN

#### Figura 4.2 Architettura della rete SpoofGAN suddivisa in generazione delle impressioni (sopra) e apposizione della texture (sotto).

SpoofGAN [53] riprende il concetto di master-fingerprint introdotto dall’algoritmo SFinGe [1], creando modelli binari di dimensioni 256x256 bit a partire da una distribuzione normale, per definire la struttura delle creste da cui verranno create varie impressioni. Le impressioni sono create con un modello standard di GAN addestrato solamente su impronte reali, per evitare artefatti e problemi di realismo dovuti alla produzione o alla presentazione delle impronte falsificate. Le deformazioni realistiche vengono introdotte in seguito alla creazione con traslazioni, rotazioni e deformazioni non lineari. Queste ultime sono calcolate tramite un modello statistico.

Come ultima operazione si appone la texture all’impronta, tramite un modello encoder-decoder che funziona come generatore, trasformando l’immagine binaria deformata in un’impressione realistica di impronta digitale. Un vettore casuale di texture viene campionato da una distribuzione normale e inserito nella generazione di texture per promuovere la diversità tra i campioni generati. Per mantenere l’identità dell’impronta e controllare la realisticità delle texture applicate, SpoofGAN utilizza due valori di perdita oltre alla normale funzione di perdita delle GAN. Un valore di perdita di identità viene calcolato tramite la distanza Euclidea tra le caratteristiche delle impressioni dello stesso dito; il secondo valore è la distanza Euclidea tra le immagini binarie originali e le immagini binarie estratte dalle foto con texture, chiamato perdita dei pixel. Questo secondo valore è calcolato sulle immagini binarie per permettere alla rete di generare diverse pressioni, umidità e contrasto nelle altre impressioni.

Grazie a questo processo SpoofGAN rende possibile l’aggiunta di nuove impronte con diversi tipi di materiali di falsificazione nel dataset, ampliando la mole di dati disponibile senza la necessità di creare l’impronta falsificata, risparmiando tempo e risorse. Inoltre aiuta il modello a sviluppare una migliore capacità di generalizzazione, ma necessita di una rete di apposizione di texture per ogni tipo, visto che il modello di encoder-decoder fornisce un miglior output se addestrato solo su di uno stile. Questo difetto lo rende computazionalmente dispendioso e non ottimale come soluzione alla problematica della molteplicità dei materiali per la falsificazione di impronte. La dipendenza dalla master-fingerprint, inoltre, limita la varianza delle immagini generate e il tipo di materiale è limitato dai materiali utilizzati nell’addestramento.

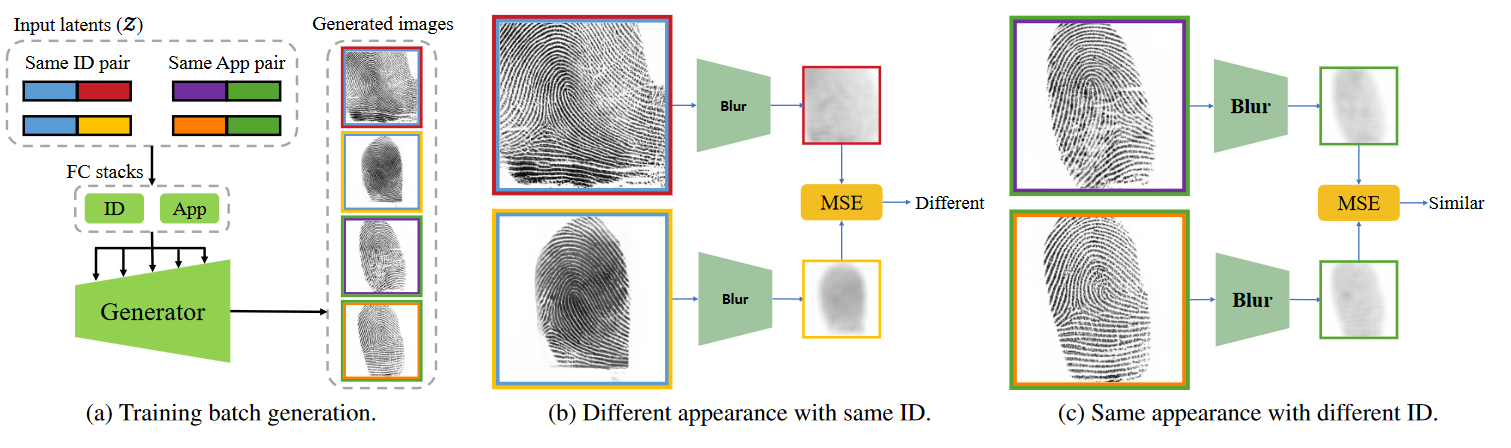


#### Figura 4.3 Immagini generate da SpoofGAN. Impressioni triplici dello stesso dito generate da SpoofGAN in cui (a) e (b) sono rappresentazioni in stile “live”, mentre (c) e (d) sono rappresentazioni in stile “falso” in gelatina e silicone (body double).

### 4.1.2 FPGAN-Control

FingerPrint GAN-Control [20] si basa su una variazione delle GAN chiamata GAN-Control [52], che permette la variazione di attributi delle immagini create, sviluppato per la creazione di volti.

GAN-Control utilizza una suddivisione dello spazio delle feature latenti in N+1 sotto-spazi, in cui N indica il numero di attributi da controllare, tranne l’ultimo sotto-spazio, che codifica gli attributi non controllabili. Uno spazio latente è una rappresentazione vettoriale di dimensioni inferiori rispetto allo spazio delle feature dell’immagine, in cui sono codificate le informazioni più importanti. Viene utilizzata una seconda GAN che permette il controllo esplicito del valore dei sottospazi, rendendo possibile variare feature come l’angolazione o caratteristiche estetiche del viso. FPGAN-Control (Figura 4.4) è l’applicazione di questo framework al campo delle impronte digitali. In FPGAN-Control non è necessario il controllo manuale delle caratteristiche, quindi sono state implementate solo la suddivisione degli spazi latenti per separare identità e apparenza delle impronte, che vengono codificate in due sottospazi differenti. [20] definisce l’apparenza delle impronte come le caratteristiche non legate all’identità dell’impronta, come tipo, metodo di acquisizione, umidità ed altro. Sono creati due sotto-spazi rappresentati da sotto-vettori, ognuno di dimensioni 256 bit e per cui viene allocato un Multi Layer Perceptron (MLP) con 8 strati. FPGAN-Control utilizza una funzione di perdita normalizzata basata sulla distanza tra identità (calcolata tramite distanza del coseno) e apparenza (calcolata tramite distanza tra due immagini nel dominio delle basse frequenze), insieme ad un peso per la perdita di apparenza, che permette di variare quanto simili debbano essere le immagini sintetiche alle immagini originali. Grazie al peso per la perdita di apparenza è possibile scegliere di quanto variare l’apparenza di impronte con la stessa identità, consentendo un’alta variabilità intraclasse, oppure variare impronte con diversa identità fino ad ottenere un’immagine quasi identica in apparenza ma con diversa identità. I modelli ResNet [37] e MobileNet [71] addestrati con dataset sintetici creati con FPGAN-Control hanno raggiunto risultati equivalenti, se non superiori, ai modelli addestrati soltanto con dati reali (Tabella 4.1). Lo spazio di possibili stili di impronte generate resta comunque limitato dalla distribuzione di stili del training test originale e la mancanza di un metodo di controllo diretto delle feature non permette di apporre materiali differenti non presenti nel dataset iniziale alle impressioni create.



#### Figura 4.4 Schema della rete FPGAN-Control proposta da Shoshan et al.

#### Tabella 4.1 Risultati degli sperimenti di Shoshan et al. [20] delle performance di varie reti addestrate sul dataset NIST SD 302 standard, augmentato con altri modelli basati su GAN e augmentato con FPGAN-Control con diversi valori di perdita di apparenza.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Training dataset** | **Res18** | **Res34** | **Res50** | **Res101** | **Mob-050** | **Mob-100** | **Eff-s** |
| Real data | 90.54 | 89.72 | 92.00 | 93.69 | 93.74 | 94.22 | 93.69 |
| StyleGAN2 | 23.54 | 6.085 | 6.47 | 8.87 | 30.70 | 32.37 | 5.59 |
| StyleGAN | 69.61 | 63.09 | 71.65 | 72.26 | 73.99 | 79.09 | 67.83 |
| FPGC-0 | 87.53 | 87.25 | 87.01 | 87.60 | 88.72 | 90.13 | 83.87 |
| FPGC-0.25 | 86.13 | 85.83 | 87.74 | 89.01 | 88.13 | 90.75 | 80.40 |
| FPGC-0.5 | 88.00 | 87.24 | 86.71 | 87.26 | 88.20 | 90.50 | 80.72 |
| FPGC-1 | 89.42 | 87.70 | 88.60 | 89.51 | **90.63** | **91.14** | 85.20 |
| FPGC-5 | 87.48 | 86.97 | 87.81 | 90.15 | 88.72 | 89.68 | 82.82 |
| FPGC-20 | **89.57** | **89.72** | **90.19** | **91.08** | 89.99 | 90.88 | **88.55** |
| FPGC-0.25 +FPGC-20 | 91.60 | 91.59 | 91.99 | 90.70 | 920.53 | 93.22 | 88.96 |
| FPGC-0.5 + FPGC-20 | **92.15** | **92.24** | 92.47 | 91.36 | **92.62** | 92.78 | **90.60** |
| FPGC-1 + FPGC-20 | 91.86 | 91.88 | **92.58** | 87.33 | 91.98 | **93.23** | 89.88 |
| FPGC-5 + FPGC-20 | 91.22 | 91.06 | 91.88 | **91.69** | 91.51 | 92.21 | 89.76 |

### 4.1.3 Universal Material Generator

Universal Material Generator (UMG) è un modello sofisticato basato sullo Style Transfer proposto da Chugh et al. [40] nel 2019, che si pone come obiettivo il miglioramento delle capacità di generalizzazione tra materiali e tra sensori dei PAD. Il metodo proposto utilizza una rete convoluzionale MobileNet-V1 come classificatore, a cui vengono fornite immagini reali e sintetiche prodotte dall’involucro UMG (Figura 4.5). Il modulo ha la funzione di generare impronte falsificate su materiali sconosciuti, trasferendo le caratteristiche stilistiche del materiale sconosciuto su impronte realizzate in materiali noti. Per fare questo utilizza un’architettura mista tra encoder-decoder e GAN. Il primo passo della rete è l’encoding di un’immagine falsa x e dello stile y di una seconda immagine tramite un encoder formato dai primi livelli del modello convoluzionale VGG-19 preaddestrato, con i pesi congelati. Si ottiene così una rappresentazione in uno spazio latente delle caratteristiche dell’immagine. Per il trasferimento di stile viene utilizzato un modulo di normalizzazione di istanza adattivo (AdaIN) [75] che riceve in input il primo dato x ed il secondo stile y e utilizza trasformazioni affini per allineare la media e la varianza dei canali di x con media e varianza di y nello spazio latente. AdaIN calcola in modo adattivo i parametri affini dallo stile in input, senza impararli dai dati come fatto dagli altri metodi di normalizzazione di istanza, tramite l’equazione:

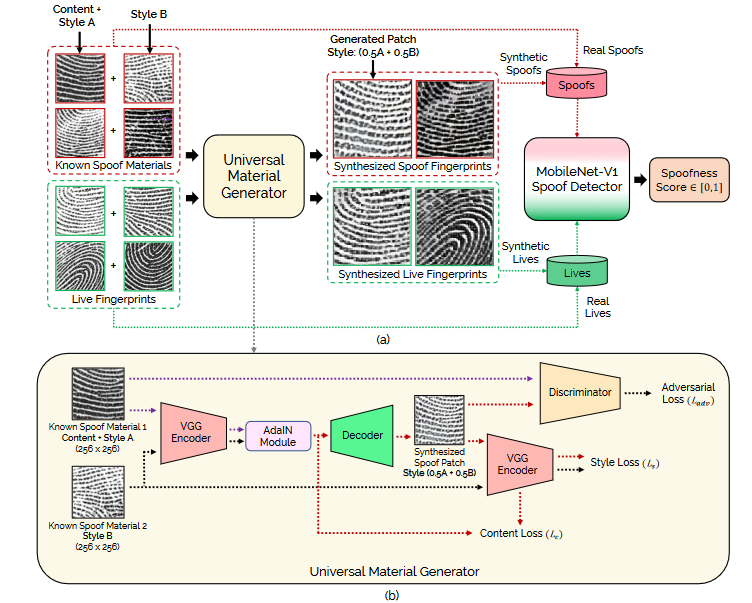
AdaIN codifica così lo stile y nello spazio latente delle variabili dell’impronta in input x, per poi costruire l’immagine risultante (x con lo stile y) con un decoder, che esegue all’inverso le operazioni convoluzionali dell’encoder VGG-19. Per assicurarsi che lo stile ottenuto sia uguale a quello scelto viene utilizzata una perdita di stile calcolata tra l’immagine ottenuta e lo spazio delle feature in ogni livello dell’encoder VGG-19. La perdita di stile viene calcolata come:

in cui *φi* rappresenta uno strato della rete, *t* l’input della rete, *g(t)* l’output del decoder e *s* rappresenta lo stile da apporre.

La magnitudine del trasferimento di stile può essere calcolato tramite l’interpolazione delle mappe delle caratteristiche. Per controllare che lo schema delle creste non sia differente dall’immagine di partenza viene utilizzata una perdita di contenuto, calcolata come la distanza Euclidea tra le caratteristiche dell’immagine generata e quelle dell’immagine iniziale, ma questa non è sufficiente. Per ovviare a questa problematica viene utilizzata la funzione di perdita avversaria tipica delle GAN con l’aggiunta dei due valori di perdita precedentemente descritti, utilizzati come pesi. In UMG l’encoder-decoder funge da generatore ed un discriminatore deve distinguere le immagini generate da quelle originarie.

Dato un dataset di immagini reali contenenti rilevazioni falsificate in vari materiali, dal dataset viene separato un materiale, che sarà considerato un materiale sconosciuto, dal resto dei materiali, che saranno considerati materiali conosciuti. Le immagini di materiali conosciuti sono utilizzate per addestrare il modulo UMG. Vengono successivamente utilizzate immagini di materiali conosciuti a coppie per generare un dataset di immagini sintetiche, che vengono immesse nel dataset originario delle falsificazioni e utilizzate per addestrare il PAD. Il processo di addestramento di UMG e immissione nel dataset originario viene fatto anche per le immagini live, in modo che venga bilanciata la distribuzione dei dati e il classificatore non si basi su eventuale rumore creato dal modulo UMG.

Il modulo UMG può essere integrato in qualunque PAD per renderlo più robusto, [40] utilizza UMG applicato al metodo Fingerprint Spoof Buster [6] per testare l’algoritmo sul dataset MSU-FPAD-v2 e LivDet 2017. Il più grande vantaggio è l’incremento della capacità di generalizzazione, che si mostra nel riconoscimento di attacchi di presentazione con materiali non conosciuti dal modello. Per testare questo, sul dataset MSU FPAD-v2 è stato utilizzato il protocollo leave-one-out, in cui, su 12 materiali, il modello viene ripetutamente addestrato su 11 materiali e testato sul materiale sconosciuto per ogni materiale presente nel dataset. I risultati sono visibili nella Tabella 4.2. Il dataset LivDet 2017 è suddiviso in modo che nel test set siano presenti solo materiali (gelatina, latex e silicone liquido) non presenti nella parte di addestramento, che presenta solo campioni in colla per legno, silicone e gomma siliconica. Inoltre LivDet 2017 contiene rilevazioni ottenute con 3 sensori differenti, perciò per testare la generalizzazione in base al sensore le reti sono state addestrate con 2 sensori noti e testate sulle rilevazioni del sensore sconosciuto. Le performance della rete sono sensibilmente aumentate sia nella generalizzazione per materiali che per sensore, come visibile nella Tabella 4.3.



#### Figura 4.5 Struttura generale della rete (a) e struttura di UMG (b) come proposto da Chugh et al. [40].

#### Tabella 4.2 Capacità di generalizzazione, calcolata come APCER con BPCER allo 0.2%, in base ai materiali del metodo Spoof Buster con e senza il modulo UMG, sul dataset MSU FPAD-v2. Il modello Spoof Buster con modulo UMG risulta superiore per tutti i materiali.

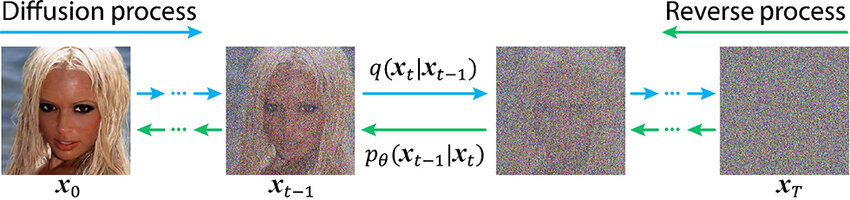
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Materiale sconosciuto** | **N° Immagini** | **N° Patch Locali** | **Performance di generalizzazione**  **APCER (%) @ BPCER = 0.2%** | |
| **Fingerprint** **Spoof** **Buster [6]** | **FSB + modulo UMG** |
| **Silicone** | 1,160 | 38,145 | 67,62 | 98.64 |
| **Latex Liquido Monster** | 882 | 27,458 | 94. 77 | 96.24 |
| **Plastillina** | 715 | 17,602 | 58.42 | 72.36 |
| **Carta stampata 2D** | 481 | 7,381 | 55.44 | 80.22 |
| **Colla per legno** | 397 | 12,681 | 86.38 | 98.97 |
| **Dita d’oro** | 295 | 9,402 | 88.22 | 88.59 |
| **Gelatina** | 294 | 10,508 | 54.95 | 97.96 |
| **Gomma siliconica curata** | 285 | 7,700 | 97.48 | 100 |
| **Vernice per corpo (lattice)** | 176 | 6,366 | 76.35 | 89.72 |
| **Trasparente** | 137 | 3,846 | 95.83 | 100 |
| **Inchiostro conduttivo su carta** | 50 | 2,205 | 90.00 | 100 |
| **Silicone 3D Universal** | 40 | 1,085 | 95.00 | 100 |
| **Totale Spoof** | **4,912** | **144,379** | Media pesata \* (± deviazione standard pesata) | |
| **Totale Live** | **5,743** | **228,143** | **75.24 ± 15.21** | **91.7 ± 9.43** |

#### Tabella 4.3 Capacità di generalizzazione in base a materiali e sensori del metodo Spoof Buster con e senza il modulo UMG, sul dataset LivDet 2017. Il modello Spoof Buster con modulo UMG risulta superiore per tutti i sensori.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **LivDet 2017** | | **Fingeprint Spoof Buster [6]** | | **Fingerprint Spoof Buster+ UMG wrapper** | |
| **Sensor in Training** | **Sensor in Testing** | **ACE (%)** | **APCER @ BPCER = 1.0 %** | **ACE (%)** | **APCER @ BPCER = 1.0%** |
| Green Bit | Orcanthus | 49.43 | 0.00 | **66.05** | **21.52** |
| Green Bit | Digital Persona | 89.37 | 57.48 | **94.81** | **72.91** |
| Orcanthus | Green Bit | 69.93 | 8.00 | **81.75** | **30.91** |
| Orcanthus | Digital Persona | 57.99 | 4.97 | **76.36** | **28.46** |
| Digital Persona | Green Bit | 89.54 | 57.06 | **96.35** | **85.21** |
| Digital Persona | Orcanthus | 49.32 | 0.00 | **68.44** | **20.38** |
| Media + dev. standard | | 67.60 ± 18.53 | 21.25 ± 28.07 | **80.63** ± **12.88** | **43.23** ± **28.13** |

## 4.2 Modelli probabilistici di diffusione

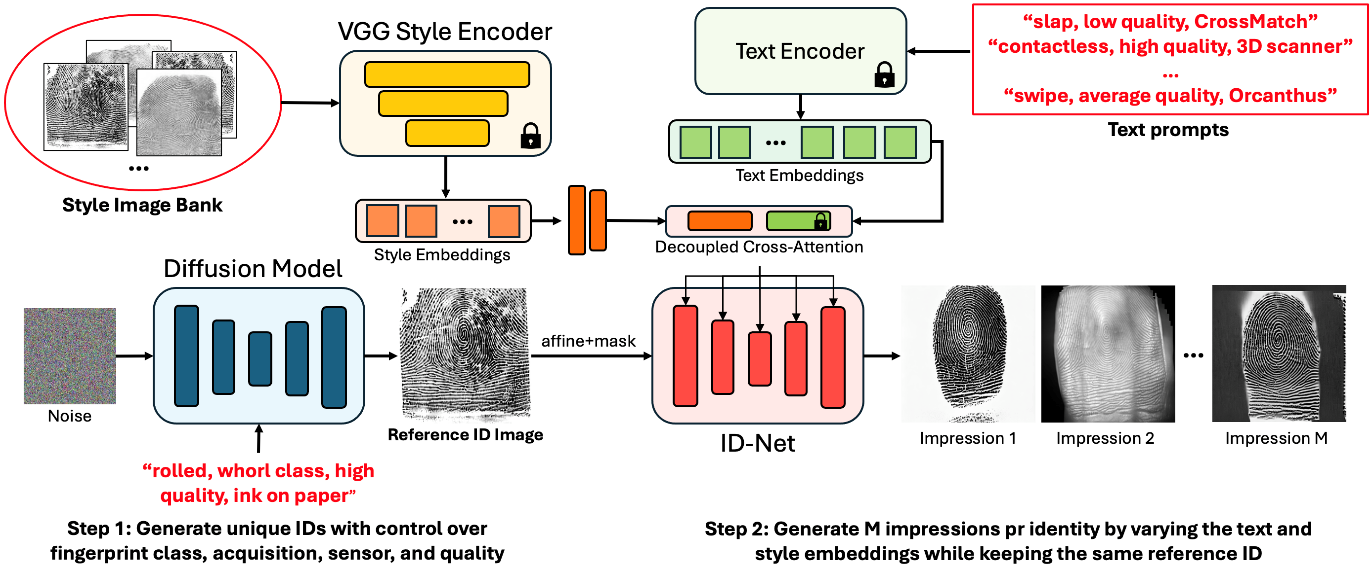
I modelli probabilistici di diffusione (DDPM) sono stati introdotti nel 2015 [70] e resi famosi da Ho et al. [69] nel 2020. Questi modelli hanno rapidamente superato le capacità delle GAN nella generazione di immagini e video in qualità, realismo e durata. Le reti DDPM sono basate sul processo stocastico markoviano, un processo aleatorio in un sistema in cui la probabilità di transizione ad un altro stato dipende solo dallo stato precedente del sistema, per convertire una distribuzione semplice in una distribuzione target (Figura 4.6). Questo processo viene chiamato diffusione e compone il primo passaggio della rete, in cui viene aggiunto gradualmente del rumore Gaussiano ad una distribuzione di dati, fino a convertirla in una distribuzione Gaussiana. Il processo generativo, che compone il secondo stadio della rete, consiste nella graduale rimozione di rumore della distribuzione Gaussiana precedentemente ottenuta seguendo i passaggi di diffusione all’inverso, ottenendo così una distribuzione di dati che segue la distribuzione originale. A causa del suo funzionamento, il secondo stadio viene chiamato processo inverso. La funzione di perdita utilizzata da questi modelli calcola la differenza tra la distribuzione di probabilità originale e quella ottenuta, utilizzando la divergenza di Kullback-Leibler. I modelli DDPM necessitano di un dataset esteso e hanno costi computazionali elevati, ma permettono un controllo maggiore sul processo generativo rispetto alle soluzioni precedenti.



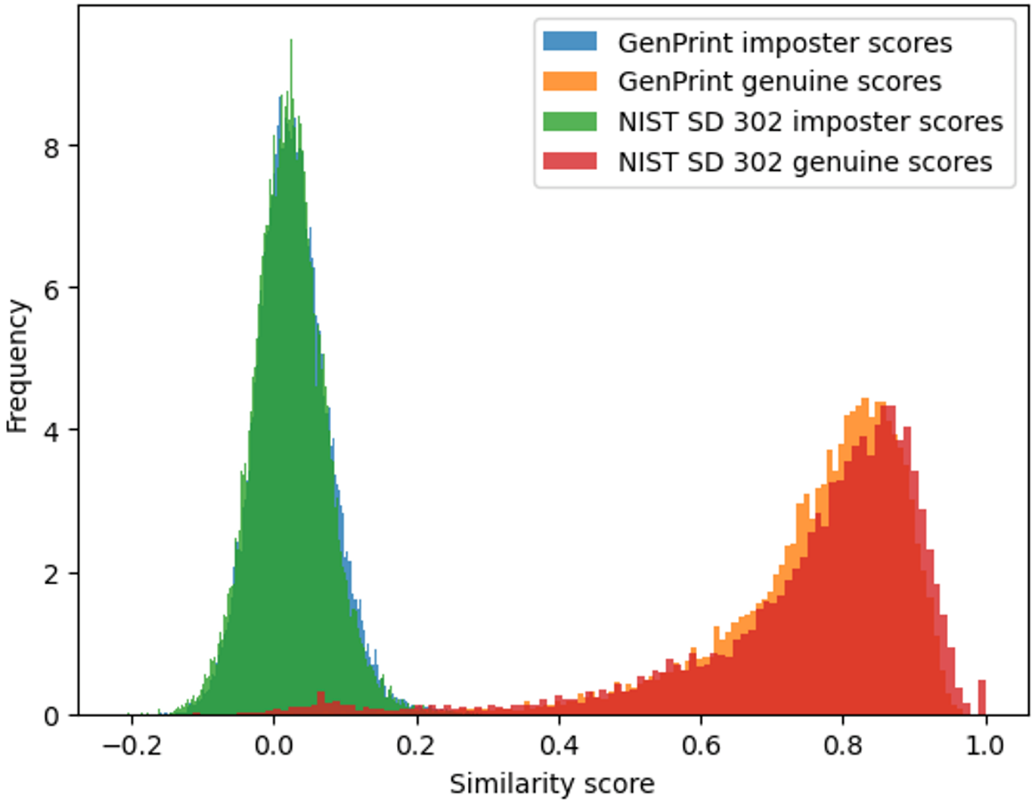
#### Figura 4.6 Funzionamento di una rete DDPM. Passando da una distribuzione iniziale x0 ad una target xt si effettua il processo di diffusione, poi invertendo il processo viene creata una nuova distribuzione che segue quella iniziale x0.

### 4.2.1 GenPrint

GenPrint [21] (Figura 4.7) è un modello a diffusione probabilistica multimodale controllabile proposto nel 2024, capace di creare immagini zero-shot, cioè senza avere ricevuto in training esempi specifici dell’immagine da generare. GenPrint permette di creare impressioni di impronte digitali da testo, tramite un modello Stable Diffusion [72] rifinito per il compito. Il dataset utilizzato per questo compito è un’aggregazione di vari dataset etichettati ottenuti con diversi sensori di rilevamento, stili di impronte e qualità differenti. Il prompt testuale permette di scegliere tipo di acquisizione, classe, qualità e sensore dell’immagine da generare. Data la difficoltà nell’esprimere le differenze intra-classe (le differenze tra rilevazioni della stessa impronta) in un prompt testuale, viene utilizzato un modello VGG preaddestrato sul dataset ImageNet per creare degli embedding, cioè rappresentazioni numeriche dense e a bassa dimensione di dati complessi, per gli stili delle immagini di addestramento. Questi embedding vengono inseriti nel modello tramite dei livelli di attenzione incrociata distaccati da quelli degli embedding testuali. Durante l’inferenza, questi embedding degli stili di vari sensori presenti nel dataset di addestramento possono essere campionati per creare immagini con stile simile alle rilevazioni di quei sensori. Può essere estratto un embedding dello stile di un sensore non visto in training, rendendo il modello capace di generazione zero-shot. Il modello utilizza una versione adattata di ControlNet, chiamata ID-Net [73], per preservare l’identità delle impronte, inserendo un modulo per l’estrazione dello schema delle creste dell’immagine in input (SqueezeUNet). Questo modulo rimuove i fattori dipendenti dal sensore e altre caratteristiche di stile, lasciando solo la silhouette del pattern delle creste e preservando sia l’identità dell’impronta che posizione e orientamento delle minutiae. La creazione delle immagini è divisa in due stadi: nel primo stadio vengono generate diverse impronte dal modello Stable Diffusion rifinito, a partire da un vettore di rumore casuale, creando uno schema delle creste utilizzato per generare variazioni intra-classe tramite variazioni di stile controllabili. Variando il vettore di rumore per ogni esecuzione vengono creati nuovi schemi unici. Nel secondo stadio le immagini generate vengono passate a ID-Net che ne modifica l’apparenza tramite gli embeddings dello stile e differenti prompt testuali che forniscono il sensore di acquisizione, la qualità ed il tipo di acquisizione. Infine viene applicata una maschera per evitare problemi di incongruenza tra l’immagine di input e di output (come per esempio il differente stile di acquisizione), che possono creare immagini non realistiche, e una griglia di distorsioni realistiche non lineari per ogni generazione. La realisticità delle immagini generate da GenPrint è stata testata tramite un classificatore ARF-net [74] per il matching di impronte, utilizzando un dataset sintetico creato da GenPrint e il dataset reale NIST SD 302 per calcolare il punteggio di impressioni vere e sintetiche, ottenendo ottimi risultati (Figura 4.8). Nella comparazione delle statistiche tra le impronte reali e quelle prodotte da GenPrint, si è visto che le immagini sintetiche sono più vicine a quelle reali di quelle prodotte da altre metodologie e migliorano l’addestramento dei sistemi di riconoscimento utilizzati. Nonostante ciò vi è ancora una differenza sia nel numero medio di minutiae che nel numero di pixel delle acquisizioni. Il vantaggio maggiore si ottiene nei dataset arricchiti con GenPrint, che hanno raggiunto performance superiori, in alcuni casi anche del 10%, per ogni stile di impronta, nell’autenticazione dell’identità. Questo metodo permette di aggiungere molta più variabilità al dataset rispetto ai metodi precedentemente utilizzati, aggiungendo sia nuovi campioni che differenti rilevazioni di questi. La possibilità di creare rilevazioni di sensori non presenti nel dataset fornisce uno strumento per la risoluzione della dipendenza dal sensore, ma è comunque limitato ad impressioni di immagini reali e non falsificate, per cui rimangono le problematiche legate ai materiali.



#### Figura 4.7 Architettura di GenPrint come proposta da Grosz et al. [21]



#### Figura 4.8 Distribuzioni dei punteggi di impostori e genuini del dataset iniziale e delle impronte create con GenPrint. La sovrapposizione delle distribuzioni mostra quanto siano accurate e realistiche le immagini generate.

## 4.4 Data Augmentation e generazione sintetica a confronto

Lo sviluppo di tecniche di Data Augmentation si è mescolato con lo sviluppo delle tecniche di generazione sintetica con l’avvento delle reti GAN, capaci sia di migliorare le qualità delle immagini, sia di creare nuovi campioni. Durante l’evoluzione delle nuove tecniche di Intelligenza Artificiale la ricerca nel campo del FPAD si è più indirizzata verso le tecniche generative come soluzione alla necessità di raccolta dei dati e delle problematiche legate alla privacy. Inoltre i modelli di diffusione e Style Transfer dimostrano una maggiore realisticità delle immagini, con un minor numero di artefatti e allucinazioni rispetto alle GAN. Tra le caratteristiche che hanno portato questi sviluppi è molto importante la differenza della quantità di dati ottenibili con i metodi di Data Augmentation e i metodi di generazione sintetica.

FPGAN-Control [20] è stato utilizzato per creare fino a 80000 nuovi campioni, partendo da 2000 impressioni nel dataset NIST SD302, con una distanza di apparenza dagli originali di 0.058 e migliorando sia la diversità del dataset sia le performance delle reti addestrate su questo (Tabella 4.1). Universal Material Generator [40] è in grado di creare un numero molto maggiore di campioni, in differenti materiali. A partire dal dataset MSU FPAD v2, composto da 10655 campioni, di cui 5743 live e 4912 falsi, sono state create 144379 patch false e 228143 patch live. Queste patch possono essere interpolate tra di loro per creare nuovi campioni in materiali differenti, ottenendo un numero di campioni altamente realistici irraggiungibile dai metodi di Data Augmentation, sebbene ad un costo e tempo computazionale più alto. I modelli di diffusione potrebbero essere un utile strumento per la generazione sintetica ma al momento non esistono modelli in grado di variare il materiale dell’impronta. In conclusione, i metodi di generazione sintetica basati sulle GAN hanno superato ampiamente le capacità di miglioramento di un dataset dei metodi di Data Augmentation, sia quantitativamente che qualitativamente, essendo più resistenti alla qualità delle immagini, alle differenze nei sensori e ai materiali sconosciuti.

# Capitolo 5

# Conclusioni

I sistemi biometrici per il rilevamento di impronte digitali sono ampiamente utilizzati in vari settori, come le banche, l’amministrazione pubblica e gli smartphone. La vulnerabilità di questi sistemi agli attacchi è una problematica reale che richiede l’integrazione di metodi di protezione adeguati. Questa tesi ha affrontato le principali sfide legate alla raccolta e all'utilizzo di dataset nei sistemi di rilevamento degli attacchi di presentazione (PAD) basati su impronte digitali. Poiché le capacità di generalizzare efficacemente dei modelli PAD allo stato dell’arte dipende fortemente dalla qualità e dalla quantità di dati disponibili, l’obbiettivo di questo studio è stato quello di approfondire le tecniche più avanzate di Data Augmentation e generazione sintetica.

I metodi per il miglioramento dei dataset noti come Data Augmentation sono ampiamente utilizzati nel campo dell’intelligenza artificiale, ma solo alcuni di questi sono utilizzabili per i dataset di impronte digitali e comprendono le trasformazioni geometriche di rotazione, specchiamento ed ingrandimento, l’aggiunta di rumore e le trasformazioni fotometriche. Queste metodologie presentano serie limitazioni, tra cui la dipendenza dalla distribuzione dei dati originaria, la dipendenza dalla qualità delle rilevazioni e la realisticità delle immagini create. A causa di queste limitazioni e dell’impossibilità di replicare le caratteristiche di nuovi materiali utilizzati dagli attaccanti, la sola Data Augmentation non è sufficiente per risolvere i problemi di cui oggi soffrono i dataset per sistemi PAD.

Con l’avvento delle reti neurali profonde sono stati introdotti modelli basati su GAN, Style Transfer e diffusione di rumore. Questi modelli sono in grado di produrre molti più campioni con una maggiore realisticità, ma anche questi sono soggetti ad errori e allucinazioni. Le GAN sono state il ponte che ha portato dalla Data Augmentation alla generazione sintetica di dati, che permette di introdurre nel dataset un grande numero di campioni con differenti proprietà rispetto alla distribuzione dei dati iniziale, migliorando le prestazioni dei sistemi PAD. I metodi di generazione più evoluti permettono di risolvere la problematica dei materiali sconosciuti dal modello e dei materiali teoricamente infiniti, producendo un altissimo numero di campioni con differenti caratteristiche, mantenendo l’identità legata all’impronta.

L’analisi di queste tecniche ha permesso di concludere che le metodologie di Data Augmentation possono fornire dei miglioramenti ad un basso costo computazionale, ma, non essendo sufficienti, per ovviare alle problematiche dei dataset di impronte è necessario implementare metodi di generazione sintetica. Il modello Universal Material Generator è risultato il metodo di generazione sintetica più adatto al miglioramento dei dataset di impronte. UMG è basato su GAN e Style Transfer ed è in grado di simulare rilevazioni di sensori e materiali mai visti, consentendo una maggiore resilienza sia ai differenti materiali che ai diversi sensori di rilevamento.

# Bibliografia

[1] Cappelli, Raffaele, Dario Maio, and Davide Maltoni. "Sfinge (synthetic fingerprint generator)." (2004).

[2] Petroc Taylor.” Global smartphone fingerprint sensor penetration rate 2014-2018“. https://www.statista.com/statistics/804269/global-smartphone-fingerprint-sensor-penetration-rate/ . (2023).

[3] Rui, Zhang, and Zheng Yan. "A survey on biometric authentication: Toward secure and privacy-preserving identification." IEEE access 7 (2018): 5994-6009.

[4] Casula, Roberto, et al. "Are spoofs from latent fingerprints a real threat for the best state-of-art liveness detectors?." 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021.

[5] Nogueira, Rodrigo Frassetto, Roberto de Alencar Lotufo, and Rubens Campos Machado. "Fingerprint liveness detection using convolutional neural networks." IEEE transactions on information forensics and security 11.6 (2016): 1206-1213.

[6] Chugh, Tarang, Kai Cao, and Anil K. Jain. "Fingerprint spoof buster: Use of minutiae-centered patches." IEEE Transactions on Information Forensics and Security 13.9 (2018): 2190-2202.

[7] Matsumoto, Tsutomu, et al. "Impact of artificial" gummy" fingers on fingerprint systems." Optical security and counterfeit deterrence techniques IV. Vol. 4677. SPIE, 2002.

[8] Collins, Clarence Gerald. Fingerprint science: How to roll, classify, file, and use fingerprints. Custom Publishing Company, 1985.

[9] Abdullayeva, Fargana, et al. "Analysis of security vulnerabilities in biometric systems." The second international conference: problems of cybernetics and informatics. 2008.

[10] Galbally, Javier, et al. "Introduction to presentation attack detection in fingerprint biometrics." Handbook of Biometric Anti-Spoofing: Presentation Attack Detection and Vulnerability Assessment. Singapore: Springer Nature Singapore, 2023. 3-15.

[11] Van de Water, Marjorie. "Can fingerprints be forged?." The Science News-Letter 29.774 (1936): 90-92.

[12] Wehde, Albert, and John Nicholas Beffel. Finger-prints can be forged. Tremonia Publishing Company, 1924.

[13] El-Abed, Mohamad, and Christophe Charrier. "Evaluation of biometric systems." New Trends and Developments in Biometrics (2012): pp-149.

[14] Li, Zewen, et al. "A survey of convolutional neural networks: analysis, applications, and prospects." IEEE transactions on neural networks and learning systems 33.12 (2021): 6999-7019.

[15] Rui, Zhang, and Zheng Yan. "A survey on biometric authentication: Toward secure and privacy-preserving identification." IEEE access 7 (2018): 5994-6009.

[16] Chugh, Tarang, Kai Cao, and Anil K. Jain. "Fingerprint spoof buster." arXiv preprint arXiv:1712.04489 (2017).

[17] Sousedik, Ctirad, and Christoph Busch. "Presentation attack detection methods for fingerprint recognition systems: a survey." Iet Biometrics 3.4 (2014): 219-233.

[18] Sorzano, Carlos Oscar Sánchez, Javier Vargas, and A. Pascual Montano. "A survey of dimensionality reduction techniques." arXiv preprint arXiv:1403.2877 (2014).

[19] Wyzykowski, André Brasil Vieira, Mauricio Pamplona Segundo, and Rubisley de Paula Lemes. "Level three synthetic fingerprint generation." 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021.

[20] Shoshan, Alon, et al. "FPGAN-Control: A controllable fingerprint generator for training with synthetic data." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2024.

[21] Grosz, Steven A., and Anil K. Jain. "Universal Fingerprint Generation: Controllable Diffusion Model with Multimodal Conditions." arXiv preprint arXiv:2404.13791 (2024).

[22] ISO/IEC 30107-3:2023 Information technology — Biometric presentation attack detection — Part 3: Testing and reporting,2023

[23] Micheletto, Marco, et al. "LivDet2023-fingerprint liveness detection competition: advancing generalization." 2023 IEEE International Joint Conference on Biometrics (IJCB). IEEE, 2023.

[24] http://biometrics.cse.msu.edu/Publications/Databases/MSU\_FPAD

[25] Mura, Valerio, et al. "LivDet 2017 fingerprint liveness detection competition 2017." 2018 international conference on biometrics (ICB). IEEE, 2018.

[26] Jawade, Bhavin, et al. "GestSpoof: Gesture Based Spatio-Temporal Representation Learning for Robust Fingerprint Presentation Attack Detection." 2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG). IEEE, 2024.

[27] Yu, Yirong, et al. "A review of fingerprint sensors: Mechanism, characteristics, and applications." Micromachines 14.6 (2023): 1253.

[28] Schneider, John K. "Ultrasonic fingerprint sensors." Advances in biometrics: Sensors, algorithms and systems. London: Springer London, 2008. 63-74.

[29] Han, Hiro, and Yasuhiro Koshimoto. "Characteristics of thermal-type fingerprint sensor." Biometric Technology for Human Identification V. Vol. 6944. SPIE, 2008.

[30] Klauk, Hagen. "Organic thin-film transistors." Chemical Society Reviews 39.7 (2010): 2643-2666.

[31] Spinoulas, Leonidas, et al. "Multi-modal fingerprint presentation attack detection: Evaluation on a new dataset." IEEE Transactions on Biometrics, Behavior, and Identity Science 3.3 (2021): 347-364

[32] Ghiani, Luca, et al. "Fingerprint liveness detection using local texture features." *IET Biometrics* 6.3 (2017): 224-231.

[33] Ojala, Timo, Matti Pietikainen, and Topi Maenpaa. "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns." *IEEE Transactions on pattern analysis and machine intelligence* 24.7 (2002): 971-987.

[34] Ojansivu, Ville, and Janne Heikkilä. "Blur insensitive texture classification using local phase quantization." *Image and Signal Processing: 3rd International Conference, ICISP 2008. Cherbourg-Octeville, France, July 1-3, 2008. Proceedings 3*. Springer Berlin Heidelberg, 2008.

[35] Hasanpour, Seyyed Hossein, et al. "Lets keep it simple, using simple architectures to outperform deeper and more complex architectures." *arXiv preprint arXiv:1608.06037* (2016).

[36] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." *The journal of machine learning research* 15.1 (2014): 1929-1958.

[37] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[38] Pan, Sinno Jialin, and Qiang Yang. "A survey on transfer learning." *IEEE Transactions on knowledge and data engineering* 22.10 (2009): 1345-1359.

[39] Carneiro, Romulo Ferrer L., et al. "Techniques of binarization, thinning and feature extraction applied to a fingerprint system." *International Journal of Computer Applications* 103.10 (2014): 4.

[40] Chugh, Tarang, and Anil K. Jain. "Fingerprint spoof generalization." *arXiv preprint arXiv:1912.02710* (2019).

[41] Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "Image style transfer using convolutional neural networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[42] Vaswani, A. "Attention is all you need." *Advances in Neural Information Processing Systems* (2017).

[43] Liu, Zicheng, et al. "Automix: Unveiling the power of mixup for stronger classifiers." *European Conference on Computer Vision*. Cham: Springer Nature Switzerland, 2022.

[44] Takahashi, Ryo, Takashi Matsubara, and Kuniaki Uehara. "Ricap: Random image cropping and patching data augmentation for deep cnns." *Asian conference on machine learning*. PMLR, 2018.

[45] DeVries, Terrance. "Improved Regularization of Convolutional Neural Networks with Cutout." *arXiv preprint arXiv:1708.04552* (2017).

[46] Zhong, Zhun, et al. "Random erasing data augmentation." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. No. 07. 2020.

[47] Yang, Zhen, et al. "Region-aware random erasing." *2019 IEEE 19th International Conference on Communication Technology (ICCT)*. IEEE, 2019.

[48] Yun, Sangdoo, et al. "Cutmix: Regularization strategy to train strong classifiers with localizable features." *Proceedings of the IEEE/CVF international conference on computer vision*. 2019.

[49] Walawalkar, Devesh, et al. "Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification." *arXiv preprint arXiv:2003.13048* (2020).

[50] French, Geoff, Avital Oliver, and Tim Salimans. "Milking cowmask for semi-supervised image classification." *arXiv preprint arXiv:2003.12022* (2020).

[51] Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems* 27 (2014).

[52] Shoshan, Alon, et al. "Gan-control: Explicitly controllable gans." *Proceedings of the IEEE/CVF international conference on computer vision*. 2021.

[53] Grosz, Steven A., and Anil K. Jain. "Spoofgan: Synthetic fingerprint spoof images." *IEEE Transactions on Information Forensics and Security* 18 (2022): 730-743.

[54] Lim, Sungbin, et al. "Fast autoaugment." *Advances in neural information processing systems* 32 (2019).

[55] Cubuk, Ekin D., et al. "Autoaugment: Learning augmentation strategies from data." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019.

[56] Cubuk, Ekin D., et al. "Randaugment: Practical automated data augmentation with a reduced search space." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020.

[57] Ho, Daniel, et al. "Population based augmentation: Efficient learning of augmentation policy schedules." *International conference on machine learning*. PMLR, 2019.

[58] Jeong, Jongheon, et al. "Smoothmix: Training confidence-calibrated smoothed classifiers for certified robustness." *Advances in Neural Information Processing Systems* 34 (2021): 30153-30168.

[59] Zhang, Hongyi. "mixup: Beyond empirical risk minimization." *arXiv preprint arXiv:1710.09412* (2017).

[60] Kim, Jang-Hyun, et al. "Co-mixup: Saliency guided joint mixup with supermodular diversity." *arXiv preprint arXiv:2102.03065* (2021).

[61] Kim, Jang-Hyun, Wonho Choo, and Hyun Oh Song. "Puzzle mix: Exploiting saliency and local statistics for optimal mixup." *International conference on machine learning*. PMLR, 2020.

[62] Chou, Hsin-Ping, et al. "Remix: rebalanced mixup." *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*. Springer International Publishing, 2020.

[63] Hendrycks, Dan, et al. "Augmix: A simple data processing method to improve robustness and uncertainty." *arXiv preprint arXiv:1912.02781* (2019).

[64] Nishiyama, Masashi, et al. "Sensation-based photo cropping." *Proceedings of the 17th ACM international conference on Multimedia*. 2009.

[65] Li, Debang, et al. "A2-RL: Aesthetics aware reinforcement learning for image cropping." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[66] Chen, Pengguang, et al. "Gridmask data augmentation." *arXiv preprint arXiv:2001.04086* (2020).

[67] Hwang, Hyunha, Kyujoong Lee, and Hyuk-Jae Lee. "Improving Corruption Robustness with Random Erasing in the Frequency Domain." *2023 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE, 2023.

[68] Kumar Singh, Krishna, and Yong Jae Lee. "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization." *Proceedings of the IEEE international conference on computer vision*. 2017.

[69] Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." *Advances in neural information processing systems* 33 (2020): 6840-6851.

[70] Sohl-Dickstein, Jascha, et al. "Deep unsupervised learning using nonequilibrium thermodynamics." *International conference on machine learning*. PMLR, 2015.

[71] Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[72] Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[73] Gao, Feng, et al. "ID-Net: an improved mask R-CNN model for intrusion detection under power grid surveillance." *Neural Computing and Applications* 33 (2021): 9241-9257.

[74] Xu, Chunbo, et al. "Arf-net: An adaptive receptive field network for breast mass segmentation in whole mammograms and ultrasound images." *Biomedical Signal Processing and Control* 71 (2022): 103178.

[75] Huang, Xun, and Serge Belongie. "Arbitrary style transfer in real-time with adaptive instance normalization." *Proceedings of the IEEE international conference on computer vision*. 2017.

[76] Johnson, Peter, Fang Hua, and Stephanie Schuckers. "Texture modeling for synthetic fingerprint generation." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2013.

[77] Larkin, Kieran G., and Peter A. Fletcher. "A coherent framework for fingerprint analysis: are fingerprints holograms?." *Optics express* 15.14 (2007): 8667-8677.

[78] Cao, Kai, and Anil K. Jain. "Learning fingerprint reconstruction: From minutiae to image." *IEEE Transactions on information forensics and security* 10.1 (2014): 104-117.

[79] Neurotechnology Inc, “Verifinger,” http://www.neurotechnology.com/verifinger.html

[80] Jain, A.K., Ross, A.A., Nandakumar, K. (2011). Introduction. In: Introduction to Biometrics. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-77326-1\_1

[81] Alsaadi, Israa Majeed. "Study on most popular behavioral biometrics, advantages, disadvantages and recent applications: A review." *Int. J. Sci. Technol. Res* 10.1 (2021).

[82] Harris Hawthorne Wilder and Bert Wentworth. *Personal Identification: Methods for the Identification of Individuals, Living Or Dead*. Boston: Richard G. Badger, The Gorham Press, 1918, 139. Public domain.

[83] Vacca, John R. *Biometric technologies and verification systems*. Elsevier, 2007.

[84] Yager, Neil, and Adnan Amin. "Fingerprint classification: a review." *Pattern Analysis and Applications* 7 (2004): 77-93.

[85] Galbally, Javier, Julián Fierrez, and Javier Ortega-García. "Vulnerabilities in biometric systems: Attacks and recent advances in liveness detection." *Database* 1.3 (2007): 1-8.

[86] Jeong, Jongkil Jay, et al. "User Characteristics and Their Impact on the Perceived Usable Security of Physical Authentication Devices." *IEEE Transactions on Human-Machine Systems* (2024).

[87] Wang, Zaitian, et al. "A Comprehensive Survey on Data Augmentation." *arXiv preprint arXiv:2405.09591* (2024).

[88] Alzubaidi, L., Zhang, J., Humaidi, A.J. *et al.* Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J Big Data* **8**, 53 (2021). <https://doi.org/10.1186/s40537-021-00444-8>

[89] Tran, Linh Duy, Son Minh Nguyen, and Masayuki Arai. "GAN-based noise model for denoising real images." *Proceedings of the Asian Conference on Computer Vision*. 2020.

[90] Jiang, Yulei, et al. "A study of the effect of noise injection on the training of artificial neural networks." *2009 International Joint Conference on Neural Networks*. IEEE, 2009.

[91] Masashi Nishiyama, Takahiro Okabe, Yoichi Sato, and Imari Sato. 2009. Sensation-based photo cropping. In Proceedings

of the 17th ACM international conference on Multimedia. 669–672.

[92] Taylor, Luke, and Geoff Nitschke. "Improving deep learning with generic data augmentation." *2018 IEEE symposium series on computational intelligence (SSCI)*. IEEE, 2018.

[93] Dalrymple, B.E. (2006). Fingerprints. In: Mozayani, A., Noziglia, C. (eds) The Forensic Laboratory Handbook. Forensic Science and Medicine. Humana Press. <https://doi.org/10.1385/1-59259-946-X:117>