DOMANDE

1. **Haar Cascade Classifiers**: come funzionano?
   Un Haar Classifier, o Haar cascade classifier, è un programma di rilevamento di oggetti di machine learning che identifica gli oggetti in un'immagine e in un video.
   Si Inizia con il calcolo delle **Haar Features**: questi sono dei weak classifiers basati su features semplici (rettangolari). Sono elementi rettangolari che possono essere divisi verticalmente in chiaro e scuro oppure divisi orizzontalmente in chiaro e scuro o con pattern misti. Ogni feature si trova in una sottoregione di una sottofinestra dell'immagine (finestra scorrevole sull'immagine). Le feature vengono applicate modificandone le dimensioni, la forma e la posizione nella finestra secondaria. Il **valore restituito** dal sistema per queste Haar Features è la somma dei valori dei pixel che rientrano nell'area bianca meno la somma dei valori dei pixel che rientrano nell'area nera.
   Possiamo stabilire un **threshold** per questa differenza in modo che possa in qualche modo restituire un risultato sì/no in base alla presenza di questo tipo di pattern nell'immagine originale.
   Questo tipo di calcolo può essere **impegnativo** dal punto di vista computazionale. Quindi usiamo le **Integral Images** per trovare un modo per calcolare rapidamente il valore di quella differenza.
   Possiamo usare **AdaBoost** per selezionare un piccolo sottoinsieme di tutte le possibili features per costruire un buon classificatore. Questa è una tecnica di allenamento che ha lo scopo di apprendere la migliore sequenza di weak classifiers e i loro pesi corrispondenti. AdaBoost apprende una sequenza di weak classifiers e li combina riducendo al minimo il limite superiore per l'errore di classificazione. L'ideale sarebbe ridurre al minimo il risultato per ogni passaggio scartando il maggior numero possibile di regioni non facciali e continuando con il minor numero di possibili regioni con immagine vera da sottoporre al classificatore successivo. Il problema è che, una volta scartata una regione, questa non verrà più considerata e restituiremo direttamente un risultato parziale senza volto per quella regione dell'immagine. I risultati positivi del primo classificatore attivano la valutazione di un secondo classificatore (più complesso) e così via. Un esito negativo in qualsiasi momento porta al rifiuto immediato della sottofinestra; una volta rifiutata, una certa sottofinestra in un certo punto verrà etichettata come “non-faccia” senza passare al classificatore successivo. Solo le finestre secondarie dell'immagine che hanno eseguito correttamente la catena complessiva verranno classificate come “faccia”.
   Ogni classificatore si è addestrato sui falsi positivi delle fasi precedenti. La localizzazione dei volti avviene analizzando sottofinestre consecutive (sovrapposte) dell'immagine come input e valutando per ognuna di esse se appartiene alla classe dei volti.

2. Perché preferiamo un basso **False Acceptance Rate** rispetto a un basso **False Rejection Rate**?
   Ci concentriamo principalmente sul **False Acceptance** quando abbiamo un'applicazione relativa alla sicurezza; questo perché, mentre il **False Rejection** può essere solo fastidioso e un po' frustrante per l'utente, il False Acceptance può essere pericoloso in molte situazioni perché, ad esempio, può consentire l'accesso a un terrorista o a un criminale.
   Se diamo allarmi per un false rejection, possiamo creare panico, quindi anche FR è importante. In generale, FA è la situazione più critica ma la criticità dipende dal tipo di applicazioni.

3. Quali sono le buone regole per il **training**?
   Quando scegliamo il **training set**, devono essere inclusi in esso modelli di diversa qualità, perché deve includere il maggior numero possibile di condizioni diverse che si troveranno nel test set o nella vita reale.
   Il Dataset deve considerare le **distorsioni** (**PIE**: Pose, Illumination, Expression) e avere buoni campioni positivi e buoni campioni negativi (qualcosa che sembra un volto ma non è un volto).
   I **Positive Examples** aiutano il sistema ad apprendere quali sono gli elementi stereotipati che compongono un volto (nel caso del rilevamento del volto). I **Negative Examples** devono essere almeno tanti quanti quelli Positivi; questi non sono campioni che contengono l'oggetto che stiamo cercando ma possono ingannare il sistema. Quindi sono campioni che potrebbero essere classificati erroneamente. Dobbiamo sottolineare l'addestramento del sistema facendo in modo che il sistema impari dalle immagini che possono ingannarlo.

4. Soft e strong biometric traits. Quali sono le differenze?
   Il **tratto biometrico** è tutto ciò che riguarda l'aspetto di una persona o il suo comportamento. Alcuni tratti sono più ‘difficili' di altri: per essere un **Strong Biometric Trait** che possa essere utilizzato qualunque sia il nostro ambiente o qualunque sia il tipo di utente, ci sono alcune caratteristiche che devono essere rispettate.
   Un strong biometric trait è un tratto biometrico universale (**Universalità**), quindi deve essere posseduto da qualsiasi persona (tranne rare eccezioni). Tutte le persone che eventualmente entrano nel sistema devono possedere questo tratto.
   Deve essere un tratto unico (**Unicità**). L'unicità di un tratto biometrico significa che ogni coppia di persone dovrebbe essere diversa in base a quel tratto.
   Un altro requisito importante per un strong biometric trait è la **Permanenza** (non deve cambiare nel tempo).
   La **Collectability** è importante perché non possiamo utilizzare il tratto biometrico, soprattutto su larga scala, se è difficilmente collezionabile. Quindi il tratto biometrico dovrebbe essere misurabile da alcuni sensori.
   Infine c'è l'**Accettabilità**: le persone coinvolte non dovrebbero avere obiezioni a consentire la raccolta/misurazione del tratto.
   Possiamo avere anche un **Soft Biometric Trait**: è un tratto che può essere utilizzato sia per tagliare la ricerca che per migliorare l'accuratezza del sistema;

può essere utile ma manca di alcune delle funzionalità precedenti. Si ottengono rilassando l'unicità; possiamo ricavare dei tratti biometrici che pur non essendo utili per riconoscere la singola persona, possono essere utili per riconoscere un gruppo di persone per ridurre la ricerca.
I Soft Biometric Traits mancano di unicità (ad es. colore dei capelli) o di permanenza (ad es. comportamenti che sono influenzati dall'umore, dalla salute, ecc.).

5. Cosa manca ai tratti biometrici comportamentali?
   I **tratti comportamentali** o **Behavioral traits** sono difficili da riprodurre ma soffrono di **permanenza**. Esempi di caratteristiche comportamentali sono la biometria delle firme e la digitazione delle chiavi.

6. Verification system e EER (Equal Error Rate)
   Nel **Face Recognition** acquisiamo ed elaboriamo i dati biometrici dell'utente al fine di restituire una decisione di autenticazione basata sull'esito di un processo di matching del modello memorizzato con quello corrente (nel caso della verifica si ha un corrispondenza 1:1; nel caso dell'identificazione si ha un corrispondenza 1:N). Nella **Verifica** una persona rivendica un'identità e presenta il suo sample come probe; il sistema ha in archivio uno o più dei suoi template; abbina questo o quei modelli con il modello in entrata; prima di decidere se quella persona è chi ha affermato di essere, il sistema confronta la somiglianza rispetto all'abbinamento con l'acceptance threshold (tale soglia viene decisa in anticipo, in fase di progettazione del sistema); se la misura di similarity soddisfa la soglia di accettazione, l'identità rivendicata è accettata; altrimenti è respinta come impostore.
   Le misure più comuni per confrontare i Sistemi di Verifica sono: tracciare la curva prodotta dal **False Acceptance Rate** con la curva prodotta dal **False Rejection Rate**; un importante punto operativo che prende il nome di **Equal Error Rate** (**ERR**, è dato da quella soglia dove il FAR è più o meno uguale al FRR); la **Receiving Operating Characteristic Curve** (**ROC**) che traccia una sorta di comportamento generale del sistema (ha FAR sull'asse x e 1-FRR sull'asse y; più alta è la curva verso la metà superiore sinistra del quadrante delle coordinate, migliore è il sistema). Il punto fornito dalla soglia in cui FAR e FRR hanno raggiunto un valore simile è chiamato Equal Error Rate (EER). L'EER non è una soglia ma è il valore che abbiamo raggiunto impostando una soglia di similarità dove il FAR è uguale al FRR.

7. Come misurare FA (False Acceptance) nella verifica?
   Quando un soggetto impostore viene accettato, abbiamo un **False Accept**, FA (è indicato anche come False Match, FM o errore di tipo II). Ciò è dovuto all'eccessiva flessibilità del sistema che può portare ad accettare la persona che non è chi afferma di essere. Il **False Acceptance Rate** (**FAR**) è definito come la percentuale di casi di identificazione in cui si verifica un False Acceptance. Tuttavia dobbiamo prendere in considerazione il numero di impostori che hanno inviato i loro probe al sistema; non quanti probe abbiamo elaborato dal sistema.

Abbiamo in questo caso **due scenari**: in uno scenario il probe appartiene ad un soggetto non registrato; nell'altro scenario il probe può appartenente anche ad utente iscritto. Questo perché anche un utente autentico può dichiarare un'identità sbagliata.

8. FN (False Negative) e TN (True Negative). I numeri assoluti FN e TN non bastano, perché?
   Quando l'identità dichiarata è vera ma il soggetto viene rifiutato, abbiamo un **False Rejection**, FR, o **False Negative**, FN. In questo abbiamo un'affermazione genuina, quindi la persona è chi afferma di essere, ma il sistema non raggiunge un punteggio sufficiente, quindi o la somiglianza è troppo bassa o la distanza è troppo alta rispetto al acceptance threshold.
   Quando un soggetto impostore viene rifiutato, abbiamo un **Genuine Reject**, GR, or **True Negative**, TN. La persona rivendica un'identità diversa da quella vera e il sistema fornisce correttamente un valore di confronto che ci fa rifiutare l'affermazione.
   Misure come FAR, FRR, CMS, ... non sono sufficienti per dare una valutazione approfondita degli algoritmi.
   Per un **confronto affidabile** dei sistemi dobbiamo considerare: il numero e le caratteristiche dei database utilizzati; dimensione delle immagini (immagini più grandi possono rappresentare una risoluzione maggiore); dimensione del Probe e della Gallery (in particolare la dimensione relativa); quantità e qualità delle variazioni indirizzate e tollerate; possibile **interoperabilità** (ad es. generalizzazione tra set di dati).

9. Cos'è la ROC curve?
   La **Receiving Operating Characteristic (ROC) curve** è una delle misure più comuni per confrontare i sistemi di verifica. La ROC curve traccia una sorta di comportamento generale del sistema.
   La ROC curve rappresenta la probabilità di Genuine Accept Rate (GAR) del sistema, espressa come 1-FRR, rispetto alla variazione di False Accept Rate (FAR). Più alta è la curva verso la metà superiore sinistra del quadrante delle coordinate, migliore è il sistema. Abbiamo un intervallo da 0 a 1.
   Potrebbe essere difficile confrontare i sistemi in base alle curve, quindi possiamo utilizzare l'area sotto la curva ROC come valore unico per il confronto.

10. Quali sono le differenze tra l'Identification Open Set e l'Identification Closed Set?
    Nell'**identificazione** non vi è alcuna rivendicazione di identità. Prendiamo ogni probe e le confrontiamo con la gallery complessiva. Quindi dal punto di vista dell'identità, abbiamo una corrispondenza 1-N.
    Esistono due tipi di identificazione: Open Set e Closed Set.
    Nell'**Identificazione Open Set** il sistema determina se il probe appartiene a un soggetto in galleria. Il sistema determina innanzitutto se l'individuo è iscritto alla galleria. Dopo di che vogliamo ottenere l'identità di quella persona.
    La differenza con la verifica è che l'individuo non fa alcuna richiesta di identità. In questo caso si possono avere più possibili situazioni di errore, perché

dipendono dal matcher e dalla soglia di riconoscimento.
Il threshold determina principalmente se il probe in ingresso viene riconosciuto come probe appartenente alla gallery.
Dato che dobbiamo effettuare confronti da 1 a n, quindi confrontiamo la sonda con tutti i template di tutte le identità presenti nella gallery, otteniamo una **lista di valori** che viene ordinata secondo i criteri che abbiamo adottato. Nel senso che se abbiamo ad esempio delle somiglianze, abbiamo una lista di valori in ordine decrescente.
Quando abbiamo un rilevamento corretto, quindi quando almeno uno dei valori nella lista supera la soglia di accettazione (nel caso della similarity), possiamo anche considerare la posizione nell'elenco in cui viene restituito il primo modello per l'identità corretta (perché l'identità corretta potrebbe non essere nella prima posizione). Possiamo considerare quindi il suo rank (rango), cioè la posizione nella lista del template appartenente all'identità corretta.
Il **Detection and Identification Rate** (**DIR**) **al rango k**, misura la probabilità di una corretta identificazione al rango k (il soggetto corretto viene riportato almeno nelle prime k posizioni).
Nell'**Identificazione Closed Set** è un caso speciale dell'identificazione dell'insieme aperto in cui si assume che ogni probe appartiene a un soggetto iscritto. In questo cosa non abbiamo nessun threshold.
L'unico tipo di errore che possiamo fare è restituire l'identità sbagliata, perché quella persona sarà sicuramente nella lista. Quindi c'è solo una sorta di False Rejection (quando restituiamo la giusta identità in una posizione che non è la prima) e **non c'è nessun False Acceptance**.
L'Identificazione Closed Set non è un'impostazione molto realistica ma è utile per valutare le prestazioni del sistema, proprio perché la valutazione dell'Identificazione Open Set è più difficile da calcolare.
Per valutare le prestazioni di un sistema di Identificazione Closed Set creiamo la cosiddetta **Cumulative Match Characteristic** (**CMC**) curve.
La probabilità di avere l'identificazione corretta all'interno della posizione k è chiamata **Cumulative Match Score** (**CMS**) al rango k.
La Cumulative Match Characteristic curve traccia i valori di CMS per tutti i possibili ranghi. In generale, tutti "i ranghi possibili" significa che possiamo arrivare a calcolare la probabilità che l'identità corretta sia presente entro la fine dell'elenco, quindi raggiungiamo sicuramente la probabilità uguale ad 1. Questo perché si assume che tutte le persone siano nella gallery quindi avremo sicuramente identità corretta almeno come l'ultima identità ritornata.
Quello che ci interessa di più è CMS al rango 1, cioè qual è la probabilità di ottenere l'identità corretta esattamente nella prima posizione. Questo valore speciale è anche noto come **Recognition Rate**.

11. L'Identification open set è più difficile della verifica?
    L'attività di identificazione open set è molto più difficile per i sistemi biometrici (ma anche per gli operatori umani) rispetto all'attività di verifica, perché dobbiamo soddisfare sia l'acceptance threshold che restituire la persona corretta come identità riconosciuta.

12. Fingerprints: quali sono i diversi livelli?
    Abbiamo tre “features levels”. Il primo è quello formato dalle **Macro-Singularities**. Sono anche chiamate "**first level features**" e comportano una considerazione globale del modello di cresta. Si basano sul numero di “**deltas**”: spirali (2 delta), loop (1 delta) e archi (nessun delta).
    Il secondo livello, detto “**second level feature**”, è formato dalle “**Minutiae**”. Queste sono chiamate anche **Galton features** e sono delle “**micro-singularities**” determinate dagli end-points o dalle biforcazioni delle linee di cresta. La prima classifica, sebbene utile per tagliare la ricerca, non era decisiva per un riconoscimento finale.
    Il numero di minutiae corrispondenti che si trovano in una coppia di impronte digitali da confrontare viene utilizzato come misura della distanza.
    Con un sensore ad altissima risoluzione è possibile studiare anche i **pori** lungo le creste. Sono chiamati "**third level features**".

13. Cos’è il Poincaré index?
    Il **Poincaré index** viene utilizzato nel Fingerprint Recognition per ricavare le macro-singularities.
    Una **directional map** è un campo vettoriale (una regione piena di vettori con orientamenti diversi). La nostra impronta digitale è una curva immersa in questo campo vettoriale. L'indice di Poincaré è definito come la rotazione totale dei vettori di questo campo una volta che percorriamo la curva.
    Per le curve chiuse, l'indice di Poincaré assume solo uno dei valori discreti 0°, ± 180°, ± 360°. In particolare, per quanto riguarda le singolarità delle impronte digitali: 0° significa che non ci sono singolarità; 360° denota la presenza di un vortice; 180° indica un loop; -180° denota un delta.

14. Cos’è una directional map?
    Una **directional map** (o directional image) che è una matrice discreta i cui elementi denotano l'orientamento della tangente alle linee della cresta.
    È una matrice in cui ogni elemento corrispondente al nodo [i, j], che si sovrappone all'immagine dell'impronta digitale, denota l'orientamento medio della cresta tangente o l'orientamento in un piccolo vicinato.

15. Cos’è il crossing number? A cosa serve?
    La localizzazione delle Minutiae si basa sull'analisi del **crossing number**.
    Queste è il numero di cambiamenti di colore che avvengono nelle vicinanze del pixel di cui si tiene conto in quel momento. Calcoliamo per ogni pixel questo crossing number per determinare se rappresenta una minuzia o meno.
    Se il crossing number è uguale a 2 ad esempio, il pixel rappresenterà un punto interno di una linea di una cresta. Se il crossing number corrisponde a 1, quel pixel è un punto di terminazione perché abbiamo un solo cambio di direzione. Se invece è uguale a 3, indica una biforcazione. Se è maggiore di 3, allora appartiene a una minuzia più complessa.

16. Qual è il metodo per valutare la distanza tra due minuzie?
    Il **Ridge Count** è una misura astratta della distanza tra due punti qualsiasi di un'impronta digitale.
    Il Ridge Count è il numero di creste che separano due minuzie. Questo non può essere fatto per ogni coppia di minuzie perché gli endpoint non sono abbastanza affidabili (possono essere causati dall'interruzione di una cresta a causa di una soglia errata o altro).

17. Cos'è una Latent Fingerprint?
    Le Impronte Latenti o **Latent Fingerprints** sono impronte digitali che lasciamo su qualsiasi superficie adatta quando la tocchiamo; vengono raccolti ad esempio durante le indagini utilizzando una polvere speciale.
    In questo caso dobbiamo confrontare un'impronta digitale che di solito è completa (l'impronta digitale registrata nella galleria) con un'impronta digitale latente che di solito è una frazione (frammento dell'impronta digitale completa). Quindi in una corrispondenza tra un'impronta latente con un'impronta registrata potrebbe esserci il **problema** dell'**Allineamento**.
    L'acquisizione delle impronte latenti avviene grazie a speciali polveri e a particolari tecniche che vengono utilizzate per trasferire l'impronta da una superficie su una carta speciale.

18. Cos'è il Rubber Sheet model? Quando viene usato?
    Le **coordinate polari** semplificano l'elaborazione dell'iride (le bande circolari diventano strisce orizzontali, quindi l'intero anello dell'iride diventa un rettangolo).
    L'**iris unwrapping** è il processo di trasformazione dall'iride circolare in coordinate cartesiane in un rettangolo in coordinate pseudopolari.
    Determinare il giusto centro per le coordinate polari è di fondamentale importanza, ma **la pupilla e l'iride non sono perfettamente concentriche** e **le dimensioni della pupilla possono cambiare** a causa dell'illuminazione o di condizioni patologiche (ubriachezza o droghe).
    Il **Rubber Sheet Model** è un processo di **normalizzazione** che tiene infatti conto di fattori quali la dilatazione della pupilla e la deformazione dell'iride.
    Prendendo un numero fisso di punti su ogni raggio che è contenuto tra il confine della pupilla e il confine dell'iride, è possibile normalizzare la distanza deformata.
    Il modello mappa ogni punto dell'iride in coordinate polari dove il centro delle coordinate polari è il centro della pupilla.
    Durante la trasformazione, le nuove coordinate per ogni punto sono date da una combinazione lineare tra le coordinate del contorno pupillare e quelle del contorno esterno dell'iride.

19. Perché l'iride è così unica?
    L'iride ha componenti altamente randotipici. Randotipico significa che non è affetto da eredità familiare (non c'è familiarità; nessuna eredità da parenti o genitori). Quindi l'iride è un tratto estremamente distintivo (l'iride destra è diversa da quella sinistra e anche i gemelli hanno iridi diverse).

20. Differenza tra luce infrarossa e luce visibile.
    Le due principali modalità di acquisizione sono la Visible Light e la Infrared light.
    Con la **Visible Light** abbiamo la melanina che assorbe la luce visibile, quindi possiamo vedere chiaramente i diversi colori che possono apparire nell'iride. Gli strati che compongono l'iride sono ben visibili ma l'immagine contiene informazioni rumorose sulla trama. In questo caso, possiamo avare il problema della **Riflessione**. Questa dipende dalla fonte di luce e dovrebbe essere rilevata in anticipo. La presenza di lenti a contatto o occhiali può influenzare il tipo di riflesso che può interessare l'iride. È anche importante considerare quando abbiamo a che fare con un'iride molto scura, perché in quel caso la trama dell'iride può diventare completamente invisibile.
    Con la **Infrared Light** la melanina riflette la maggior parte della luce infrarossa, quindi non abbiamo informazioni sul colore. La trama è più visibile ma richiede attrezzature speciali.
    Se adottiamo Infrared sensors risolviamo i due problemi precedenti perché è possibile evitare la maggior parte dei riflessi che si fondano nella luce visibile ed è possibile evidenziare la trama dell'iride scura.
    Tuttavia, nelle immagini nel vicino infrarosso **mancano le componenti cromatiche**. Tutte le differenze di colore vengono eliminate.
    Inoltre c'è un altro problema: i sensori vicini all'infrarosso non sono universalmente disponibili come quelli a luce visibile.

21. Quali sono i problemi che possono sorgere nel riconoscimento dell'iride?
    I problemi che possiamo avere nell'Iris Recognition sono: la Riflessione; le piccole dimensioni dell'iride; l'alta risoluzione richiesta dall'apparecchiatura; la limitata profondità di campo per cui dobbiamo prestare attenzione ai problemi di messa a fuoco e fuori fuoco; dobbiamo allinearci con l'asse ottico a meno che non usiamo alcuni accorgimenti come il cosiddetto "problema fuori asse" (se la persona guarda in un'altra direzione, l'iride non sarà esattamente al centro della sclera, ma si sposterà verso l'estremità destra o sinistra della sclera); l'eventuale presenza di occhiali o lenti a contatto.

22. Cos'è il Borda Count? Quando lo usiamo?
    Quando effettuiamo **Score Level Fusion** ogni classificatore produce una classifica di classi in base alla probabilità del pattern di appartenere a ciascuno di essi. Ciò significa che maggiore è la probabilità, maggiore sarà la posizione nell'elenco.
    Quindi, invece di restituire una class label, viene restituita una graduatoria dei candidati.
    Nel **Borda Count** le classifiche vengono poi convertite in punteggi che vengono sommati; la classe con il punteggio finale più alto è quella scelta dal multiclassificatore.

23. Che cos'è un Approccio Multibiometrico?
    I **Sistemi Multibiometrici** o anche **Insieme di Classificatori** sono una delle soluzioni proposte che possono permetterci di migliorare le prestazioni di un sistema biometrico. Quando usiamo molti più tratti biometrici è molto più difficile falsificarli, perché l'attaccante dovrebbe usare un campione falso per

ciascuno dei tratti che sono coinvolti.
Quando parliamo di sistemi multibiometrici possiamo avere **molti modi per combinare diverse risposte biometriche**. Il più ovvio è quello di unire più tratti. Questo è il puro approccio multimodale o **Multimodal Approach**.
D'altra parte, possiamo anche avere approcci in cui possiamo sfruttare istanze multiple dello stesso tratto, ovvero il **Multiple Instances**.
Un altro tipo di sistema multibiometrico è un sistema che comporta istanze ripetute, quindi il **Repeated Instances**. In questo caso oltre ad avere esattamente lo stesso tratto biometrico, ne catturiamo due campioni.
La differenza tra questi ultimi due approcci è che con il Repeated Instances abbiamo ad esempio due impronte dello stesso dito, mentre nel Multiple Instances abbiamo due impronte di due dita diverse.
Un altro approccio popolare è il **Multiple Algorithms**. Ad esempio, applichiamo LBP e i Wavelets per elaborare un'immagine del viso; gli algoritmi di matching sono adatti ai diversi metodi di estrazione delle caratteristiche che applichiamo; quindi possiamo fondere i risultati di due diversi algoritmi.
Alla fine, abbiamo il caso dello stesso tratto o di tratti diversi (ma in generale lo stesso tratto) catturati da sensori multipli, quindi il **Multiple Sensors**. Ad esempio possiamo utilizzare due sensori diversi per acquisire la stessa impronta digitale (entrambi ottici o entrambi capacitivi o uno ottico e uno capacitivo o ecc...) e poi fondere i risultati.

24. Come possiamo fondere i risultati negli approcci multibiometrici?
    Il primo approccio è **Sensor Level Fusion**. Un esempio di questo approccio lo si ha quando fondiamo informazioni da tre immagini 2D in un unico modello 3D. In questo caso la fusione viene eseguita prima dell'estrazione delle features. Le features utilizzate per il matching e per la decisione vengono estratte direttamente dal modello fuso.
    La Sensor Level Fusion è **difficile da realizzare** perché dobbiamo avere prima di tutto lo stesso tratto e il tipo di segnale che viene estratto deve essere lo stesso.
    L'approccio successivo è quello del **Feature Level Fusion**. In questo caso acquisiamo diversi campioni (uno per ogni tratto) e poi fondiamo i feature vectors che sono stati estratti. Un esempio può essere l'estrazione delle features dall'occhio destro e dall'occhio sinistro; estraiamo separatamente i due codici dell'iride e poi li fondiamo in un unico vettore che viene poi abbinato. Questo approccio può generare feature vectors di dimensioni molto grandi, quindi si è soggetti alla "**curse of dimensionality**"; di conseguenza devono essere necessariamente eseguiti delle riduzioni di dimensionalità dello spazio oppure una feature selection.
    Un altro problema è l'**inflessibilità** di questo sistema perché il matcher viene addestrato utilizzando la nozione di "vettore fuso".
    L'approccio più popolare e più flessibile è lo **Score Level Fusion**. In questo approccio ogni tratto viene elaborato da un sottosistema diverso. Quindi abbiamo le feature extraction ed i matching che avvengono in maniera separate. La fusione verrà fatta solo dopo i matching.
    In questo caso il problema si sposta solo per trovare una buona strategia di

fusione.
L'ultimo livello di fusione possibile è rappresentato dal **Decision Level Fusion**. In questo caso siamo arrivati alla decisione finale. Ad esempio, in caso di verifica potremmo avere una risposta Sì o No da ciascuno dei sistemi coinvolti e quindi questi vengono fusi utilizzando una politica di voto a maggioranza (**majority voting**).
Il **punto debole** di questo approccio è che ogni volta che qualcosa va storto, abbiamo perso tutte le informazioni che potrebbero essere utili per valutare il comportamento del sistema.

25. Negli approcci multibiometrici, quando vogliamo fondere i risultati, quali problemi possiamo riscontrare?
I punteggi di diversi matcher possono essere **non omogenei**: similarità/distanza; diversi range (es. [0,1] o [0,100]); distribuzioni diverse.
Un altro problema è dato dall'**Affidabilità** dei diversi sistemi.

26. Nello Score Level Fusion, quali sono i tre tipi di risultati che si possono fondere?
Le regole di fusione che possono essere utilizzate per lo **Score Level Fusion** sono Abstract, Rank e Measurement.
Nell'**Abstract** il classificatore restituisce un **class label** per il modello di input. Questo class label rappresenta la risposta del classificatore a una determinata operazione di matching.
Uno dei modi più utilizzati per fondere i risultati è il voto di maggioranza o **majority voting**: ogni classificatore vota per una classe e il modello viene assegnato alla classe più votata.
Nel **Rank** ogni classificatore restituisce una classifica (un **ranking**) delle classi in base alla probabilità del pattern appartenente a ciascuno di essi.
E' possibile utilizzare il **Borda Count**, dove la classifica viene poi convertita in punteggi che vengono sommati; la classe con il punteggio finale più alto è quella scelta dal multiclassificatore.
Nel **Measurement** ogni classificatore emette il proprio punteggio (**score**) di classificazione per il modello rispetto a ciascuna classe.
L'unico problema che dobbiamo risolvere è avere una misurazione dei valori che rientri nello stesso intervallo, quindi dobbiamo applicare la **normalizzazione**.
Un metodo banale è solo quello di **sommare i punteggi** e passare il risultato alla regola della fusione. C'è solo un problema che in qualche modo impedisce la piena flessibilità di questo approccio, ovvero la possibilità di modificare il tipo di soglia che sfruttiamo per ottenere il risultato finale. Sono possibili diversi metodi, tra cui somma, somma ponderata, media, prodotto, prodotto ponderato, max, min, ecc.

27. Tutti i tratti biometrici sono soggetti a spoofing?
Quelli che si basano sull'apparenza sono soggetti a spoofing. L'andatura, la digitazione dei tasti, il modo in cui firmi, ecc., no.

28. Che cos'è lo spoofing? Come si svolge? Quali sono le strategie?

Un **Spoofing attack** cerca di ingannare l'applicazione biometrica, utilizzando una copia o eseguendo un'imitazione del tratto biometrico utilizzato da quel sistema per autenticare legittimamente un utente.

Un sistema biometrico può essere attaccato in diversi punti: nei canali di trasmissione; nell'estrattore di features; nel matcher. È possibile iniettare dati su ciascuno di tali canali per forzare il risultato desiderato. È anche possibile attaccare il Database con template che non appartengono a persone iscritte. Tutti questi tipi di attacchi lungo i canali, il database e così via, possono essere considerati come **attacchi indiretti**. Difendersi da questo tipo di attacchi è una questione di sicurezza informatica.

Un altro tipo di attacchi sono gli **attacchi diretti**. In questo caso, lo scopo finale non è riconoscere la persona, ma piuttosto rilevare l'attacco in modo da poter procedere con una contromisura.

Possiamo fare una classificazione degli attacchi di spoofing: attacco di spoofing 2D e attacco di spoofing 3D.

Negli **Spoof 2D**, l'attacco viene effettuato utilizzando una superficie 2D come una foto o un video che viene riprodotto una volta che abbiamo in qualche modo registrato la persona attaccata. Quest'ultimo tipo di attacco è anche chiamato "**Replay Attack**".

Negli **Spoof 3D**, l'attacco richiede che non ci sia una superficie piana ma una superficie tridimensionale e in generale vengono utilizzate maschere per questo tipo di attacco (**Mask Attack**).

Per ogni tipo di attacco ci sono diverse **tecniche anti-spoofing** che si basano principalmente sul **liveness detection**.

In generale, quello che cerchiamo di fare, oltre a studiare il pattern di riflessione della superficie e la microtexture, è "sfidare" l'utente al fine di rilevare il tipo di reazione della faccia che abbiamo di fronte.

Abbiamo tecniche anti-spoofing che funzionano a **livello di sensore** (basato su **hardware**); in pratica sfruttano le proprietà intrinseche di un corpo vivente, compreso il tipo di riflettanza che sta producendo e segnali involontari (ad esempio micromovimenti degli occhi) che sono completamente assenti in una foto o in una maschera. Un esempio di tecnica anti-spoofing che sfrutta questo tipo di movimenti è **Eye Blink**. In effetti, il battito delle palpebre è un movimento involontario naturale che può essere rilevato per distinguere una persona reale da una foto.

D'altra parte, possiamo avere una risposta volontaria dalla cosiddetta strategia "**Challenge-Response**". In questo caso possiamo chiedere ad una persona che sta presentando un prob di fare un certo tipo di espressione o movimento. Se ciò che ci aspettavamo non accade sulla faccia mostrata al sistema, possiamo concludere che si tratta di un attacco.

Altri tipi di tecniche anti-spoofing vengono eseguite a **livello di Feature Extractor** (basate su **software**) e possono essere statiche o dinamiche. Un esempio di studio statico è lo studio della **microtexture** dell'immagine acquisita; con la dinamica studiamo il tipo di caratteristiche dinamiche che un certo movimento può produrre quando viene eseguito naturalmente.

Possiamo avere anche una Score level fusion dove possiamo fondere anti-spoofing e riconoscimento in un unico modulo oppure effettuare anticipatamente l'anti-spoofing e procedere al riconoscimento solo se l'anti-spoofing restituisce una risposta genuina.

29. Come si valuta un sistema di anti-spoofing? È possibile valutarlo insieme al riconoscimento? Come?
    Una possibilità è sfruttare lo **Spoof False Acceptance Rate** (**SFAR**): il numero degli attacchi di spoof che vengono falsamente accettati.
    Utilizziamo lo SFAR invece del FAR perché quest'ultimo è dovuto ai cosiddetti **zero effort attacks** o attacchi zero sforzo (una persona rivendica un'identità senza intraprendere alcuna azione particolare per assomigliare all'utente autentico). In uno **Spoof Scenario** invece dobbiamo considerare anche i tentativi che sono compiuti per cercare di riprodurre l'aspetto della persona attaccata.
    Si utilizza anche il **False Rejection Rate**. In questo caso il FRR non si riferisce al mancato riconoscimento di un campione, ma all'errata classificazione del campione autentico come attacco contraffatto.
    In alcuni casi possiamo **fondere il riconoscimento e l'anti-spoofing** in un unico sistema con un'unica risposta Accetta-Rifiuta. Quindi possiamo combinare un Classificatore Biometrico con un Classificatore Binario, perché un Classificatore Antispoofing non è che un algoritmo di sistema in grado di decidere se una sonda è genuina o meno.

30. Qual è la differenza tra lo Spoofing ed il Camouflage?
    Nel caso del **Camuffamento** l'attacco viene effettuato presentando al sistema un tratto biometrico per ingannarlo fingendo di non essere se stesso. Quindi in quel caso non vogliamo essere riconosciuti. Introduciamo infatti elementi estranei sul viso il processo di rilevamento potrebbe fallire.

31. Approcci anti-spoofing per il Face Recognition
    Uno degli approcci più intuitivi a 2D Print Attack è **Liveness Detection**. In pratica, la differenza essenziale tra il viso dal vivo e la fotografia è che un viso dal vivo è un oggetto completamente tridimensionale mentre una fotografia potrebbe essere considerata come una struttura planare bidimensionale. Possiamo utilizzare la struttura dal movimento per ricavare informazioni di profondità per distinguere una persona dal vivo da una foto fissa.
    È difficile stimare le informazioni di profondità quando la testa è ferma (dobbiamo chiedere un movimento, altrimenti è molto difficile effettuare un rilevamento della vita in questo modo banale) e la stima è molto sensibile al rumore e all'illuminazione (se re in un ambiente buio, questo può aiutare l'attaccante a nascondere qualche cambiamento di caratteristica rispetto a un volto reale).
    È possibile calcolare l'**Optical Flow** (tecnica che cerca di estrarre il vettore di movimento confrontando la posizione di ogni pixel in un frame e in quello successivo) sul video in ingresso per ottenere l'informazione del movimento del viso per la livens detection.

Un possibile **approccio multimodale** fonde volto-voce contro lo spoofing sfruttando il movimento delle labbra mentre si parla.
Un'altro approccio sfrutta l'analisi del battito di ciglia essendo movimento fisiologico che non si può assolutamente evitare, ovvero l'**Eye Blink**.
Un'altra possibilità si basa sul **Micro-texture analysis**. Ciò è particolarmente efficiente quando abbiamo 2D Print attacks o Photo attacks perché qualsiasi tipo di carta fotografica o carta da stampa ha un tipo di microtexture che, pur non essendo visibile agli occhi, può essere rilevata utilizzando le texture features. I volti umani e le stampe riflettono la luce in modi diversi perché un volto umano è un oggetto 3D complesso non rigido (quindi riflette la luce in direzioni diverse) mentre una fotografia è un oggetto rigido planare (diverse sfumature e riflessi speculari). Le proprietà della superficie di facce e stampe reali, ad es. pigmenti, sono diversi anche perché i pigmenti naturali sono diversi dai pigmenti dell'inchiostro. Questi ultimi contengono alcuni componenti metallici quindi questo influenza il modo in cui la luce viene riflessa. Il lavoro sfrutta modelli binari locali multiscala (LBP). La strategia utilizzata nei modelli binari locali multiscala è più o meno la stessa ma molto più semplice da adottare rispetto al banco di wavelet con frequenze diverse. I modelli binari locali multiscala vengono calcolati utilizzando finestre di dimensioni diverse. Come ulteriore vantaggio, le stesse caratteristiche di texture utilizzate per il rilevamento dello spoofing possono essere utilizzate anche per il riconoscimento facciale. I vettori nello spazio delle caratteristiche vengono quindi inviati a un classificatore SVM che determina se i modelli di micro-trama caratterizzano una persona viva o un'immagine falsa. Un altro approccio importante è l'approccio Captured-Recaptured. Dobbiamo considerare che tutte le distorsioni che sono presenti quando catturiamo un'immagine influenzano l'immagine del viso quando passa attraverso il sistema della fotocamera. Tali distorsioni vengono applicate due volte quando scattiamo la foto di una persona e poi quando la stampiamo. In pratica, quando utilizziamo una foto scattata da una foto, come può succedere quando si utilizza una foto scattata su internet, abbiamo una qualità dell'immagine molto inferiore rispetto alla cattura di un'immagine del viso reale.
I volti umani e le stampe riflettono la luce in modi diversi perché un volto umano è un oggetto 3D complesso non rigido (quindi riflette la luce in direzioni diverse) mentre una fotografia è un oggetto rigido piano (con diverse sfumature e riflessi speculari).
Per questo si sfruttano i **Multi-scale Local Binary Patterns** (**LBP**). I Multi-scale LBP vengono calcolati utilizzando finestre di dimensioni diverse. I vettori nel feature space vengono poi inviati a un **classificatore SVM** che determina se i modelli di microtexture caratterizzano una persona dal vivo o un'immagine.
Un altro approccio importante è l'approccio **Captured-Recaptured**. Dobbiamo considerare che tutte le distorsioni che sono presenti quando catturiamo un'immagine influenzano l'immagine del viso quando passa attraverso il sistema della fotocamera. Tali distorsioni vengono applicate due volte quando scattiamo la foto di una persona e poi quando la stampiamo.
Un altro approccio è la **Gaze Stability**. Esiste un algoritmo che si basa sul presupposto che la coordinazione spaziale e temporale dei movimenti dell'occhio, della testa e (possibilmente) della mano coinvolti nel compito di

seguire uno stimolo visivo è significativamente diversa quando si fa un tentativo vero e proprio rispetto a certi tipi di tentativi di spoofing.
In una strategia **challenge-response**, quando il sistema chiede al soggetto genuino di ruotare la testa, il tipo di coordinazione temporale tra il movimento degli occhi, il movimento della testa (se eventualmente coinvolti) è diverso quando se il la persona sta guardando lo schermo attraverso i fori di una maschera. Il compito richiede di fissare un semplice bersaglio che appare su uno schermo davanti all'utente. Nel caso di un attacco di spoofing fotografico, sono necessari movimenti della mano guidati visivamente per orientare la foto in modo che punti nella direzione corretta verso la sfida. Quindi la coordinazione temporale tra le fissazioni oculari e la rotazione della testa è in qualche modo influenzata da questo movimento della mano che non è presente in una presentazione genuina. L'**Optical Flow Correlation** tiene conto del movimento della testa dell'utente che cerca di autenticarsi e del movimento dello sfondo della scena, perché se si muovono insieme significa che c'è un attacco di spoofing, perché viso e sfondo si muovono insieme mentre non dovrebbero muoversi insieme. Optical Flow non funziona in due casi: un caso nel caso delle maschere, ma anche se abbiamo un'immagine del viso piegato sul viso dell'aggressore.
Un altro possibile approccio consiste nell'utilizzare l'**Image Distortion Analysis**. Si basa sul tipo di riflessi speculari prodotti da una superficie di carta stampata o da uno schermo LCD (ad esempio lo schermo di uno smartphone) durante l'acquisizione. Esiste una sorta di analisi composta che viene effettuata in base a questi diversi elementi e quindi è possibile concatenare le caratteristiche estratte da ciascuno di questi fattori di distorsione dell'immagine e addestrare il classificatore.
Un altro modo per utilizzare efficacemente lo studio delle **microtexture** è legato ai **Replay Video Attacks**. Quando eseguiamo un Replay Video Attack, mostriamo un video su un altro schermo. In questo caso esiste un tipo speciale di pattern chiamato **moiré pattern aliasing** che è causato dalla sovrapposizione di due pattern di pixel: uno dal video originale e l'altro dallo schermo su cui viene mostrato il video. È possibile rilevare questo modello speciale utilizzando Multi-scale LBP e DSFIT (questo cerca di rilevare le caratteristiche con la più alta energia nell'immagine). In pratica, ogni volta che è presente una sorta di motivo moiré, possiamo concludere che siamo in presenza di un attacco di parodia di replay.

32. Doddington zoo. Perché è stata fatta l'estensione?
    La maggior parte degli errori nel sistema può essere attribuita a una specifica classe di utenti. Il fatto che una persona appartenga a una delle cattive classi del **Doddington zoo** può pregiudicare l'affidabilità della singola operazione di riconoscimento. Doddington ha definito pecore, capre, agnelli e lupi nel contesto dei sistemi di speaker recognition.
    Una **Pecora** è una persona che produce un campione biometrico che combacia bene con gli altri della stessa persona e male con quelli di altre persone. Quindi genera meno false accepts e rejects rispetto alla media. Una Pecora raggiunge per lo più Genuine Acceptance e raramente raggiunge una False Acceptance quando rivendica un'identità diversa.

Una **Capra** è una persona che produce un campione biometrico che corrisponde male agli altri che aveva prodotto lui stesso. Questi punteggi di corrispondenza bassi implicano un false reject rate superiore alla media per le capre. Questa è una persona che ottiene un FRR superiore alla media perché questa persona non è ben riconosciuta.
Un **Agnello** è una persona che può essere facilmente impersonata. Quando il campione biometrico di tale persona viene abbinato a un campione biometrico di un'altra persona, il punteggio di corrispondenza risultante sarà superiore alla media. Di conseguenza, i false matches sono più probabili. Questo può accadere ad esempio per i bambini.
Un **Lupo** è una persona brava a imitare. Quando una tale persona presenta un campione biometrico per il confronto, ha un'alta possibilità di generare un punteggio di corrispondenza superiore alla media rispetto a un campione memorizzato di una persona diversa. Quindi abbiamo molte False Acceptances. Capre, agnelli e lupi sono definiti in termini di punteggi medi genuini o impostori di un utente.
Le nuove aggiunte sono definite in termini sia di punteggi autentici di un utente che di quelli dell'impostore.
I **Camaleonti** sembrano sempre simili agli altri. Questo è qualcosa di leggermente diverso dai Lupi: in pratica, ha un punteggio medio alto sia con se stesso, sia quando interpreta il ruolo di impostore. Quindi ricevono un punteggio di corrispondenza elevato per tutte le verifiche. Per questo motivo, i camaleonti raramente causano falsi rifiuti, ma è probabile che causino false accettazioni. Un esempio di utente che potrebbe essere un camaleonte è qualcuno che ha caratteristiche molto generiche.
I **Fantasmi** portano a punteggi bassi indipendentemente con chi vengono confrontati. Quindi i Fantasmi hanno un punteggio medio basso sia come genuini che come impostori. I fantasmi possono essere la causa dei falsi rifiuti, ma è improbabile che siano coinvolti nelle false accettazioni. Un esempio è dato in questo caso da quelle persone che hanno problemi a registrarsi al sistema. Ciò potrebbe portare a un'estrazione delle features più difficile e di conseguenza a punteggi di corrispondenza bassi per tutte le verifiche. Ad esempio una persona con una pelle scadente sulle impronte digitali può avere problemi con l'iscrizione automatica.
Le **Colombe** hanno un punteggio medio genuino alto e punteggi medi di impostori più bassi. Sono riconoscibili, combaciano bene con se stessi e male con gli altri. Le colombe sono raramente coinvolte in qualsiasi tipo di errore di verifica.
I **Vermi** sono gli utenti peggiori perché come genuini hanno un punteggio medio basso ma come impostori un punteggio medio alto. Hanno poche caratteristiche distintive, quindi pochi sono correttamente riconosciuti come se stessi, ma possono impersonare con successo altre persone.

33. Vantaggi e svantaggi del riconoscimento 3D e 2D.
Per quanto riguarda le **variazioni di posa**, mentre il riconoscimento 2D può essere influenzato da variazioni di rotazione, pitch e yaw, questo non è vero con il 3D perché possiamo testare con diverse pose sul modello 3D: può essere ruotato e la sua posa modificata in base al pitch e allo yaw in modo da poter

provare a far assumere al modello 3D la stessa posa rappresentata nell'immagine 2D. Per un motivo simile anche i problemi di **illuminazione** possono essere considerati risolti perché, allo stesso modo, possiamo ruotare il modello e anche sintetizzare illuminazioni diverse. Mentre le **espressioni** influenzano ancora il riconoscimento facciale in 3D. Se pensiamo soprattutto alle espressioni esagerate che sono quelle più difficili da catturare anche in 2D, creano comunque problemi in 3D perché possono causare una reale modifica rispetto ad un modello 3D neutro. Anche l'**invecchiamento** può influenzare la strategia 3D. Con l'avanzare dell'età, è possibile che alcuni problemi del viso si rilassino. Poiché il volume di una determinata parte anatomica può aumentare con l'età, ciò influisce anche sulla strategia 3D.
Il **make up** non influisce il modello 3D perché solo considerando le relazioni geometriche, il volume 3D determinato dal viso non cambia. La **chirurgia plastica** può influenzare i modelli 3D a seconda del tipo di peso e dell'estensione dell'intervento. L'**occlusione** influenza il 3D come influenza il 2D.
Possiamo riassumere **i pro e i contro** delle strategie 3D dicendo che: in 3D abbiamo molte più informazioni, i modelli costruiti sono molto più robusti ad una serie di distorsioni, c'è la possibilità di sintetizzare (approssimativamente) immagini 2D da pose 3D virtuali ed espressioni calcolate da un modello 3D (**PRO**); il costo dei dispositivi, il costo computazionale delle procedure, il possibile rischio che alcuni dispositivi di acquisizione possono presentare, ad esempio il laser scanner è pericoloso per l'occhio (**CON**).

34. I Morphable model
L'espressione influisce ancora sul riconoscimento facciale in 3D. E' possibile sfruttare il cosiddetto **Morphable model**, che può essere in qualche modo distorto per riprodurre qualsiasi tipo di espressione, ma la complessità computazionale aumenta in modo drammatico.
Il modello è chiamato Morphable perché possiamo modificare alcune relazioni tra i poligoni nel modello per creare aspetti diversi della stessa faccia o anche per creare facce diverse. La forma e la trama del modello generico vengono manipolate per adattarsi alle immagini catturate.
I Morphable model consentono inoltre di sintetizzare le espressioni del viso approssimando le possibili espressioni di un soggetto specifico. Questo si basa su un set di dati di facce 3D e richiede che tra le facce ci sia una piena corrispondenza (allineamento).

35. Qual è la principale differenza tra PCA e LDA?
PCA è unsupervised. Quando raccogliamo campioni da utilizzare per costruire il nuovo feature space, non sappiamo quale sia il soggetto che è stato raffigurato in ogni immagine. Raccogliamo solo un numero di campioni abbastanza rappresentativi da soggetti diversi e proviamo a creare un nuovo feature space su cui mappiamo ogni immagine della galleria e ogni probe in arrivo prima di confrontarli.
Per questo motivo **PCA è** molto più **incline alle intra-class misclassifications**, quindi è più soggetta a falsi rifiuti, nel senso che è più facile per la PCA non

essere in grado di riconoscere lo stesso individuo in una situazione diversa.
L'idea è quella di passare a **LDA** che è un metodo **supervised** perché tiene conto non solo delle raccolte di campioni ma anche di una partizione di tali campioni nelle diverse classi rappresentate.
In **Linear Discriminant Analysis** (LDA) si etichettano i campioni per ogni soggetto che partecipa al training set. Questo tipo di partizionamento aiuta ad apprendere meglio quali sono gli elementi nel nuovo spazio che meglio si riferiscono alle intra-personal variations rispetto a quelli che in realtà si riferiscono alle inter-personal variations (che sono quelle che ci interessano).