

Tesina Intelligenza Artificiale

Luca Maria Lauricella 519997

March 2023

1 Introduzione

In questo progetto verrà approfondito la tecnica di Intelligenza Artificiale della Face Detection e della Face Comparison, si vogliono quindi individuare e riconoscere i volti nelle immagini. La Face Detection è una tecnologia sempre più presente nella nostra vita quotidiana e ne risulta interessante fare un approfondimento.

Questa tecnica ha numerose applicazioni pratiche, come ad esempio la fotografia digitale, la videosorveglianza, la sicurezza informatica e l'interazione uomo-macchina. Ad esempio, la Face Detection può essere utilizzata per identificare automaticamente le persone in una foto o in un video, o per rilevare le espressioni facciali e le emozioni

In questo progetto, verrà illustrato come utilizzare Python e la libreria OpenCV per creare un modello di riconoscimento facciale in grado di rilevare i punti chiave del viso umano, come gli occhi, il naso e la bocca, e utilizzarli per identificare la presenza di un volto.

La Face Comparison risponde alla domanda “i due visi corrispondono?”. Funziona in modo molto simile al riconoscimento facciale. Parte dall’acquisizione di un’immagine che viene confrontata con altre immagini (presenti nel database o inserite in input per il paragone). La discrepanza sta nel risultato. Infatti, la Face Comparison restituisce dei valori numerici che indicano se i volti di due immagini diverse corrispondono alla stessa persona. Per farlo vengono analizzate le caratteristiche del volto e le loro proporzioni.

2 Implementazione

In questa sezione si effettua un’analisi approfondita delle implementazioni di sistemi di Face Detection, Face Comparison e Face Recognition.

In generale per implementare un sistema di Face Detection si deve:

- Raccogliere e preparare un set di dati di immagini di volti. Questi dati saranno utilizzati per addestrare un algoritmo di rilevamento dei volti;

- Scegliere un algoritmo di rilevamento dei volti. Uno dei più popolari è l'algoritmo di Viola-Jones, ma ci sono anche altri algoritmi come l'HOG (Histogram of Oriented Gradients), il CNN (Convolutional Neural Network) e il DNN (Deep Neural Network);
- Addestrare l'algoritmo di rilevamento dei volti utilizzando il set di dati di immagini di volti. Ciò può essere fatto utilizzando tecniche di apprendimento supervisionato, in cui l'algoritmo viene alimentato con esempi di volti con etichette che indicano la presenza o l'assenza di un volto;
- Testare l'algoritmo utilizzando un set di dati di immagini che non sono state utilizzate per l'addestramento. Questo aiuterà a verificare l'accuratezza dell'algoritmo di rilevamento dei volti;;
- Implementare l'algoritmo di rilevamento dei volti nel sistema. Ciò può essere fatto utilizzando librerie di computer vision e vari linguaggi di programmazione come OpenCV, MATLAB, Python attraverso le librerie offerte come OpenCV, TensorFlow, Keras, PyTorch, e altre;
- Utilizzare il sistema per rilevare i volti in immagini o video. Ciò può essere fatto fornendo l'algoritmo di rilevamento dei volti con l'immagine o il video e utilizzando le coordinate dei volti rilevati per identificarli o analizzarli ulteriormente.

L'implementazione di un sistema di Face Comparison richiede l'utilizzo di un software di elaborazione delle immagini facciali e l'adozione di alcune misure di sicurezza, di seguito sono riportati i passaggi generali per l'implementazione di un sistema di confronto facciale:

- Raccolta di una o più immagini facciali della persona che si desidera verificare o identificare. Queste immagini possono essere acquisite tramite fotocamere, scanner o dispositivi di acquisizione di immagini;
- Pre-elaborazione delle immagini: effettuare la pre-elaborazione delle immagini acquisite per migliorarne la qualità e rimuovere eventuali rumori o artefatti. Questo può includere operazioni di filtraggio, normalizzazione, segmentazione e rilevamento delle caratteristiche facciali;
- Estrazione delle caratteristiche facciali: estrarre le caratteristiche facciali dalle immagini pre-elaborate. Questo può includere l'estrazione di punti chiave come gli occhi, il naso e la bocca, nonché l'analisi della forma del viso e di altri dettagli facciali;
- Confronto facciale: confrontare le caratteristiche facciali delle immagini di riferimento con quelle dell'immagine da verificare o identificare. Questo può essere fatto utilizzando algoritmi di confronto facciale che calcolano la somiglianza tra le caratteristiche facciali delle diverse immagini;

- Valutazione del risultato: valutare il risultato del confronto facciale e determinare se le immagini confrontate appartengono alla stessa persona o meno. Questo può essere fatto utilizzando una soglia di somiglianza predefinita, sopra la quale viene ritenuto che le immagini appartengano alla stessa persona;

Per quanto riguarda la Face Recognition, è una tecnologia di Intelligenza Artificiale che utilizza algoritmi per identificare e autenticare un individuo sulla base delle sue caratteristiche facciali uniche. Ci sono vari algoritmi e tecniche utilizzate per il riconoscimento facciale, ma in generale, il processo può essere suddiviso in tre fasi principali:

- Acquisizione dell'immagine: Il sistema di riconoscimento facciale acquisisce un'immagine del volto dell'individuo che si vuole confrontare con altre immagini;
- Preprocessing dell'immagine: L'immagine acquisita viene quindi elaborata per estrarre le caratteristiche facciali distintive, come la forma degli occhi, del naso e della bocca, l'orientamento del viso e la texture della pelle. Questo processo prevede spesso la rimozione del rumore e la normalizzazione dell'immagine per garantire una rappresentazione coerente delle caratteristiche;
- Corrispondenza delle caratteristiche: In questa fase, le caratteristiche facciali estratte dall'immagine vengono confrontate con un'altra immagine nel quale è ricercato l'individuo. Se le caratteristiche facciali corrispondono a quelle di un individuo presente nella seconda immagine, il sistema riconosce l'individuo;

L'algoritmo più comune utilizzato per il riconoscimento facciale è il metodo di analisi dei componenti principali (PCA), che riduce le caratteristiche facciali a un set di componenti principali che possono essere utilizzati per confrontare le immagini facciali. Altri algoritmi utilizzati includono la discriminazione lineare (LDA), l'analisi delle onde locali (LBP) e la trasformata di Fourier (FFT).

3 Test Set

In seguito vengono mostrate le immagini che sono state utilizzate per misurare l'efficienza e i risultati degli algoritmi di Face Detection e Face Comparison. Queste immagini rappresentano gli input, appunto, dei due algoritmi.

Face Detection : Figure 1, Figure 2, Figure 3

Face Comparison : Figure 1, Figure 3, Figure 4

.



Figure 1: Volti1



Figure 2: Volti2



Figure 3: Drake



Figure 4: OtherDrake

4 Codice

In questa sezione e nella prossima vengono riportati i due algoritmi che sono stati implementati e testati con il precedente Test Set.

Si noti come per testare la funzionalità degli algoritmi sia stato utilizzato un più

ampio set di immagini ma per semplicità se ne riportano solo una parte ridotta.

Nel codice del primo algoritmo, il classificatore pre-addestrato di Viola-Jones viene caricato utilizzando il file XML 'haarcascade-frontalface-default.xml'. L'immagine di prova viene caricata utilizzando la funzione 'cv2.imread()' e viene convertita in scala di grigi utilizzando la funzione 'cv2.cvtColor()'. L'algoritmo di rilevamento dei volti viene quindi applicato all'immagine in scala di grigi utilizzando la funzione 'face-cascade.detectMultiScale()'. Infine, viene disegnato un rettangolo intorno ai volti rilevati sull'immagine originale utilizzando la funzione 'cv2.rectangle()', e l'immagine con i volti rilevati viene mostrata utilizzando la funzione 'cv2.imshow()'.

Nel codice del secondo algoritmo invece viene utilizzata la libreria face-recognition per effettuare la comparazione di due immagini passate in input tramite la funzione compare-faces

5 Risultati

In questa sezione mostriamo i risultati ottenuti con i due algoritmi. Mentre nel primo algoritmo vengono mostrati i riquadri intorno ad ogni volto riconosciuto, nel secondo verrà mostrata una stampa che risponde alla domanda 'Ci sono corrispondenze tra le due immagini'.

Come si potrà notare in seguito, nella Face Detection non sempre vengono individuati tutti i volti, infatti si sceglie un giusto size di soglia che rappresenta il numero di caratteristiche necessarie per far sì che un volto venga classificato come tale. Questo valore è un giusto compromesso che limita il numero dei falsi positivi e falsi negativi.

Le ultime due immagine mostrano l'esito (positivo/negativo) della comparazione di due immagini



Figure 5: Volti1 FD



Figure 6: Volti2 FD

```

1 import face_recognition
2
3
4 known_image = face_recognition.load_image_file("drake.jpeg")
5 unknown_image = face_recognition.load_image_file("otherdrake.jpeg")
6
7 know_encoding = face_recognition.face_encodings(known_image)[0]
8 unknow_encoding = face_recognition.face_encodings(unknown_image)[0]
9
10 result = face_recognition.compare_faces([know_encoding], unknow_encoding)
11 answer= "Si" if result[0] else "No"
12
13 print(answer)

```

Si

Figure 7: Drake/OtherDrake Comparison

```

1 import face_recognition
2
3
4 known_image = face_recognition.load_image_file("drake.jpeg")
5 unknown_image = face_recognition.load_image_file("volti.jpg")
6
7 know_encoding = face_recognition.face_encodings(known_image)[0]
8 unknow_encoding = face_recognition.face_encodings(unknown_image)[0]
9
10 result = face_recognition.compare_faces([know_encoding], unknow_encoding)
11 answer= "Si" if result[0] else "No"
12
13 print(answer)

```

No

Figure 8: Drake/Volti Comparison