Siamese Neural Network with a Contrastive Loss

Rocco Aliberti
Università degli Studi di Salerno
Dipartimento di Informatica

Abstract

L'emergenza sanitaria scatenata dal coronavirus ha fatto emergere tutti i limiti che può avere il riconoscimento facciale da quando viene utilizzata la mascherina, infatti le persone, per ridurre il contagio del virus, hanno iniziato ad indossare la mascherina, diminuendo la capacità dei sistemi, di riconoscimento facciale, di identificare un soggetto. Quindi, per vedere quanta differenza esiste, tra i volti, quando indossano la mascherina e quando non la indossano, abbiamo utilizzato le reti neurali siamesi con la perdita di contrasto. Una rete neurale siamese è una rete neurale artificiale che utilizza gli stessi pesi mentre lavora in tandem su due diversi vettori di input per calcolare vettori di output comparabili. Verranno utilizzate per confrontare tra loro immagini di volti, dove è presente la mascherina, e dove non lo è, per scoprire quanto sono differenti le immagini. I risultati ottenuti dalle reti siamesi per ogni confronto delle immagini riportano una stima di somiglianza, la quale descrive, per ogni confronto, la differenza tra i volti con e senza la mascherina. Quindi, con lo sviluppo di questo lavoro, si è riusciti a scoprire, secondo le reti neurali siamesi, tramite un valore numerico riportato dal modello, la differenza che esiste tra un volto quando indossa la mascherina, e quando non la indossa.

1 INTRODUZIONE

L'arrivo della pandemia per l'emergenza Covid ha provocato un grande cambiamento nelle vita quotidiana delle persone, cambiando le loro abitudini. Ormai non si può più uscire e comunicare senza l'uso di una mascherina, che è diventata una caratteristica particolare del nostro volto, ma, con l'uso della mascherina, i sistemi biometrici per il riconoscimento facciale hanno iniziato a mostrare dei problemi, infatti, da quando viene utilizzata, riconoscere un volto è diventato molto più complicato. Per riuscire a risolvere questo problema esiste il bisogno di capire quanta differenza esiste tra i volti quando indossano la mascherina e quando non la indossano.

L'obiettivo di questo lavoro è scoprire questa differenza. Per fare ciò si sono confrontate delle immagini di volti, prese da un dataset, con e senza la mascherina, utilizzando le reti neurali siamesi con la perdita di contrasto.

2 RELATED WORKS

Il riconoscimento facciale è la tecnologia che ha subito più cambiamenti durante la pandemia, quindi, è stata la più sviluppata negli ultimi anni per essere sempre utile e aggiornata.

Il sistema di riconoscimento facciale, dove viene spiegato in un articolo della Cornell University, dovrebbe essere in grado di rilevare automaticamente un volto in un'immagine. Ciò comporta estrarne le caratteristiche e quindi riconoscerlo, indipendentemente da illuminazione, espressione, invecchiamento e posa, il che è un compito difficile. [1]

Un sistema per il riconoscimento facciale è stato sviluppato in Cina, dove con l'utilizzo di un IA è stato consentito al governo cinese di identificare le persone anche con la mascherina. Il sistema è stato sviluppato dalla Hanwang Technology Ltd, conosciuta anche come Hanvon. L'IA ha il merito di identificare le persone con la mascherina con una precisione del 95%. [2]

Un studio sul riconoscimento facciale è stato effettuato dall'Università Politecnica delle Marche, dove sono state utilizzate le DCNN, reti neurali convoluzionali profonde, insieme all'algoritmo per il riconoscimento facciale, per fornire un'analisi e un confronto delle moderne tecnologie, dei loro vantaggi e dei loro limiti, e di facilitare la ricerca di nuove tecniche. [3]

3 SISTEMA PROPOSTO

Le reti neurali siamesi (SNN) condividono parametri e pesi tra due o più reti sorelle, ciascuna producendo vettori di incorporamento dei rispettivi input. L'aggiornamento dei parametri viene rispecchiato in entrambe le reti sorelle e viene utilizzato per trovare somiglianze tra gli input confrontando i relativi vettori di caratteristiche, infatti, le SNN utilizzano una funzione di somiglianza per confrontare due input e dare in output uno score di somiglianza.

Nell'apprendimento per similarità, le reti vengono addestrate per massimizzare il contrasto (distanza) tra incorporamenti di input di classi diverse, riducendo al minimo la distanza tra incorporamenti di classi simili.

Verranno utilizzate le SNN per realizzare il confronto tra le immagini di volti dello stesso soggetto con e senza la mascherina.

DATASETS

Per lo sviluppo di questo lavoro si è utilizzato il dataset M2FRED (Mobile Masked Face Recognition Database), un dataset di volti che include video, in the wild, di 43 soggetti, sviluppato dal BIPLab presso l'Università degli Studi di Salerno.

Dai video presenti nel dataset sono state estrapolate delle immagini per ogni soggetto, le quali formano due dataset:

- il primo dataset è composto da sole immagini di volti senza la mascherina;
- il secondo è composto da sole immagini di volti con la mascherina.

3.1 Metodologia

Inizialmente si sono addestrate le reti siamesi solo con le immagini di volti senza la mascherina, creando coppie di immagini appartenenti allo stesso soggetto, così da definire il minimo contrasto tra immagini dello stesso volto, e coppie di immagini appartenenti a

soggetti differenti, così da definire il massimo contrasto tra immagini di volti diversi.

Figure 1: Coppie di immagini usate per l'addestramento

Dopo aver addestrato le reti siamesi, si creano le coppie di immagini di volti, appartenenti allo stesso soggetto, con e senza la mascherina da far confrontare alle reti siamesi per ottenere lo score di somiglianza.

Figure 2: Coppie di immagini usata per il confronto

4 RISULTATI SPERIMENTALI

Con l'addestramento delle reti siamesi si può verificare con quale criteri vengono confrontate le immagini tra loro dalle reti siamesi. Il modello presenta:

- Una funzione dove viene definita la distanza euclidea, utilizzata per unire gli input forniti dalle immagini e inviarli alla rete siamese;

```
# Provided two tensors t1 and t2
# Euclidean distance = sqrt(sum(square(t1-t2)))
def euclidean_distance(vects):
    """Find the Euclidean distance between two vectors.

    Arguments:
        vects: List containing two tensors of same length.

    Returns:
        Tensor containing euclidean distance
        (as floating point value) between vectors.
    """

    x, y = vects
    sum_square = tf.math.reduce_sum(tf.math.square(x - y), axis=1, keepdims=True)
    return tf.math.sqrt(tf.math.maximum(sum_square, tf.keras.backend.epsilon()))


input = layers.Input((130, 130, 3))
x = tf.keras.layers.BatchNormalization()(input)
x = layers.Conv2D(4, (5, 5), activation="tanh")(x)
x = layers.AveragePooling2D(pool_size=(2, 2))(x)
x = layers.Conv2D(16, (5, 5), activation="tanh")(x)
x = layers.AveragePooling2D(pool_size=(2, 2))(x)
x = layers.Flatten()(x)

x = tf.keras.layers.BatchNormalization()(x)
x = layers.Dense(10, activation="tanh")(x)
embedding_network = keras.Model(input, x)


input_1 = layers.Input((130, 130, 3))
input_2 = layers.Input((130, 130, 3))

# As mentioned above, Siamese Network share weights between
# tower networks (sister networks). To allow this, we will use
# same embedding network for both tower networks.
tower_1 = embedding_network(input_1)
tower_2 = embedding_network(input_2)

merge_layer = layers.Lambda(euclidean_distance)([tower_1, tower_2])
normal_layer = tf.keras.layers.BatchNormalization()(merge_layer)
output_layer = layers.Dense(1, activation="sigmoid")(normal_layer)
siamese = keras.Model(inputs=[input_1, input_2], outputs=output_layer)
```

Figure 3: Distanza Euclidea

- Una funzione dove viene definita la perdita di contrasto che verrà utilizzata dalle reti siamesi per il confronto delle immagini.

```
def loss(margin=1):
    """Provides 'constrastive_loss' an enclosing scope with variable 'margin'.

  Arguments:
      margin: Integer, defines the baseline for distance for which pairs
              should be classified as dissimilar. - (default is 1).

  Returns:
      'constrastive_loss' function with data ('margin') attached.
  """

    # Contrastive loss = mean( (1-true_value) * square(prediction) +
    #                          true_value * square( max(margin-prediction, 0) ))
    def contrastive_loss(y_true, y_pred):
        """Calculates the constrastive loss.

      Arguments:
          y_true: List of labels, each label is of type float32.
          y_pred: List of predictions of same length as of y_true,
                  each label is of type float32.

      Returns:
          A tensor containing constrastive loss as floating point value.
      """

        square_pred = tf.math.square(y_pred)
        margin_square = tf.math.square(tf.math.maximum(margin - (y_pred), 0))
        return tf.math.reduce_mean(
            (1 - y_true) * square_pred + (y_true) * margin_square
        )

    return contrastive_loss
```

Figure 4: Perdita di contrasto

Dal modello delle reti neurali siamesi si può ricavare il Model Accuracy, che è il numero di classificazioni che un modello prevede correttamente diviso per il numero totale di previsioni effettuate,

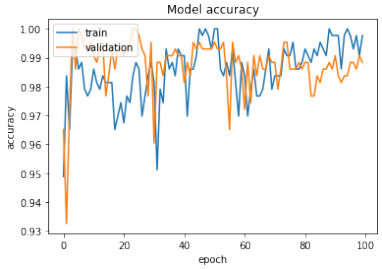

Figure 5: Grafico Model Accuracy

e il Contrastive Loss, che prende l'output della rete per un esempio positivo e calcola la sua distanza da un esempio della stessa classe e la contrasta con la distanza da esempi negativi.

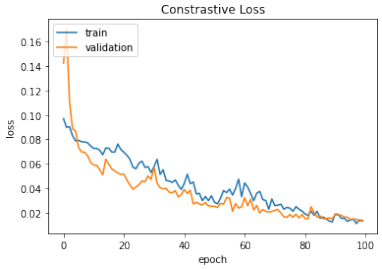

Figure 6: Grafico Contrastive Loss

Il modello è stato addestrato più volte con differenti dimensione dei lotti e epoche:

- Il modello addestrato con lotti composti da 16 coppie di immagini e 10 epoche ha il valore di Accuracy uguale al 96,9%, mentre il valore di Loss uguale al 19,9%.
- Il modello addestrato con lotti composti da 32 coppie di immagini e 10 epoche ha il valore di Accuracy uguale al 98,8% , mentre il valore di Loss uguale al 17,1%.

	Accuracy	*Loss*
Batch Size = 16 Epochs = 10	0.9698	0.1994
Batch Size = 32 Epochs = 10	0.9884	0.1717
Batch Size = 64 Epochs = 10	0.9953	0.1600
Batch Size = 16 Epochs = 100	0.9744	0.0246
Batch Size = 32 Epochs = 100	0.9744	0.0231
Batch Size = 64 Epochs = 100	0.9884	0.0140

- Il modello addestrato con lotti composti da 64 coppie di immagini e 10 epoche ha il valore di Accuracy uguale al 99,5%, mentre il valore di Loss uguale al 16%.
- Il modello addestrato con lotti composti da 16 coppie di immagini e 100 epoche ha il valore di Accuracy uguale al 97,4%, mentre il valore di Loss uguale al 0,2%.
- Il modello addestrato con lotti composti da 32 coppie di immagini e 100 epoche ha il valore di Accuracy uguale al 97,4%, mentre il valore di Loss uguale al 0,2%.
- Il modello addestrato con lotti composti da 64 coppie di immagini e 100 epoche ha il valore di Accuracy uguale al 98,8%, mentre il valore di Loss uguale al 0,1%.

In ogni tipo di addestramento si è verificato lo score di somiglianza tra le immagini, che sono state utilizzate per l'addestramento, per verificarne la validità.Per ogni coppia di immagine, dello stesso soggetto, lo score di somiglianza deve avere un valore vicino al valore 1, mentre per ogni coppia di immagine, di soggetti differenti, lo score di somiglianza deve avere un valore vicino allo 0.

Negli addestramenti effettuati con 100 epoche, lo score di somiglianza di ogni coppia di immagine, dello stesso soggetto, ha un valore molto vicino al 1, ed ogni coppia di immagine, di soggetti differenti, ha un valore molto vicino allo 0.

Figure 7: Coppie di immagini con 100 epoche

Invece, negli addestramenti effettuati con 10 epoche, lo score di somiglianza di ogni coppia di immagine, dello stesso soggetto, e di ogni coppia di immagini di soggetti differenti, hanno un valore vicino allo 0.5. Questi addestramenti non sono ottimi perchè non riescono a distinguere quando le immagini confrontate sono simili o diverse tra loro, quindi non verranno utilizzati per i confronti successivi perchè potrebbero dare dei risultati fasulli.

5 DISCUSSIONE DEI RISULTATI

Dopo aver effettuato l'addestramento dei modelli, si prendono le immagini degli stessi soggetti, con e senza la mascherina, e si danno

Figure 8: Coppie di immagini con 10 epoche

in input ai modelli, appena addestrati, per confrontarle tra loro ed avere in output lo score di somiglianza.

I risultati ottenuti dai confronti variano tra i modelli:

- per il modello addestrato con lotti di grandezza uguale a 16, gli score di somiglianza sono molto simili tra loro, infatti, si potrebbero racchiudere in 3 classi differenti, dove in ogni classe gli score di somiglianza variano intorno ad un certo valore;

Figure 9: Coppie di immagini con grandezza 16

- per il modello addestrato con lotti di grandezza uguale a 32, gli score di somiglianza sono pressochè simili tra loro, infatti, si potrebbero catalogare in 4 classi differenti, dove in ogni classe gli score di somiglianza variano intorno ad un certo valore;

Figure 10: Coppie di immagini con grandezza 32

- invece, per il modello addestrato con lotti di grandezza uguale a 64, gli score di somiglianza non sono molto simili tra loro, si potrebbero raggruppare in 6 classi differenti, dove in ogni classe gli score di somiglianza variano intorno ad un certo valore, ed è per questo che i risultati ottenuti da questo modello sembrano essere i migliori, per la varietà degli score di somiglianza che sono presenti in questo modello.

Tra i risultati ottenuti, dai confronti delle immagini, in ogni tipo di addestramento, sono presenti anche dei risultati che potrebbero essere considerati dei risultati fasulli, perché alcune immagini confrontate tra loro potrebbero essere considerate simili o diverse a causa della presenza, oltre della mascherina, anche di altri elementi che hanno alterato il risultato del confronto tra le immagini, per esempio:

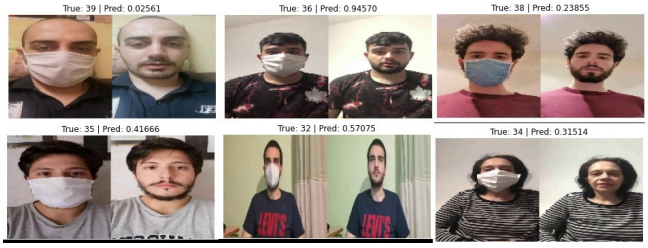

Figure 11: Coppie di immagini con grandezza 64

- la poca luminosità presente nell'immagine, le immagini potrebbero essere considerate simili tra loro perché, a causa della poca luminosità, non viene identificata la presenza della mascherina, e quindi, non viene evidenziata nessuna differenza tra le due immagini confrontate;

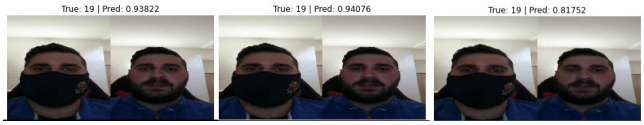

Figure 12: Immagini con bassa luminosità

- la diversa intensità di luce presente nell'immagine, le immagini potrebbero essere considerate diverse tra loro per la troppa differenza di luce;

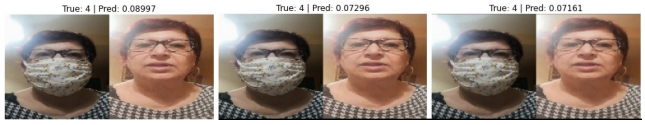

Figure 13: Immagini con differenza di luce

- la diversa postura del volto, le immagini potrebbero essere considerate diverse per la differente posizione del volto, influenzando il risultato del confronto, e quindi, i volti dello stesso soggetto potrebbero essere considerati diversi tra loro.

Figure 14: Immagini con diversa postura del volto

6 CONCLUSIONI

In questo lavoro si è riusciti a confrontare immagini di volti dove è presente la mascherina, con immagini di volti dove non lo è, grazie all'utilizzo delle SNN.

Il miglior addestramento, per realizzare i confronti tra le immagini, è quello con lotti di 64 coppie di immagini e 100 epoche, dove si è ottenuta la maggior varietà di score di somiglianza, e quindi le immagini vengono confrontate in modo più specifico a differenza degli altri modelli.

Però, la stima di somiglianza risultante dai confronti delle immagini, prodotta dalle reti neurali siamesi, potrebbe non essere sempre veritiera, perchè, si sono verificati dei risultati che potrebbero essere fasulli, in ogni tipo di addestramento, e questi risultati vengono emessi dai confronti che vengono influenzati da altri elementi presenti nell'immagine, oltre che dalla stessa presenza della mascherina, per esempio la differenza di luce, l'inclinazione della fotocamera, la postura del volto, e perciò, queste immagini potrebbero essere confrontate in modo errato dalle reti neurali siamesi.

REFERENCES

[1] Divyarajsinh N Parmar and Brijesh B Mehta. 2014. Face recognition methods & applications. *arXiv preprint arXiv:1403.0485*.

[2] Martin Pollard. 2020. Even mask-wearers can be id'd, china facial recognition firm says. *Reuters. March*, 9.

[3] FABIO D'ANGELO. 2022. Analisi dell'impatto delle foto segnaletiche nel riconoscimento di soggetti in scenari" unconstrained" mediante reti neurali profonde.