

Progetto di Fondamenti di Visione Artificiale e Biometria

Anno Accademico 2019-2020

GENDER CLASSIFICATION USING   
A SPIDER WEB METHOD

Studenti: Salvatore Froncillo 0522500858

Pasqualino Gravina 0522500864

# Sommario

[Sommario 1](#_Toc40517008)

[Introduzione 2](#_Toc40517009)

[Presentazione del progetto 2](#_Toc40517010)

[Obiettivi 3](#_Toc40517011)

[Dataset utilizzato 3](#_Toc40517012)

[Configurazioni assegnate 5](#_Toc40517013)

[Preprocessing 6](#_Toc40517014)

[Algoritmo Ragnatela 6](#_Toc40517015)

[Algoritmo Ragnatela 2.0 6](#_Toc40517016)

[Classificatori 8](#_Toc40517017)

[Classificatore SVM 8](#_Toc40517018)

[Reti neurali 9](#_Toc40517019)

[Rete neurale con vettore ragnatela 9](#_Toc40517020)

[Rete neurale convoluzionale 10](#_Toc40517021)

[Rete neurale ibrida 11](#_Toc40517022)

[Ulteriori Evidenze (dataset CelebA) 12](#_Toc40517023)

[Dataset CelebA 12](#_Toc40517024)

[“Re-allenamento” modello 12](#_Toc40517025)

[Test Incrociati 12](#_Toc40517026)

[Conclusioni 13](#_Toc40517027)

# Introduzione

## Presentazione del progetto

Immagine che contiene persona, fotografia, posando, indossando

Descrizione generata automaticamenteLa *gender classification è* una tipologia di classificazione ampiamente utilizzata, in particolare viene utilizzata per supportare diverse biometrie di riconoscimento. Tale tipo di classificazione risulta essere molto utile, ad esempio, in ambito forense l’identificazione del genere di un soggetto mediante biometria può restringere il campo di ricerca circa del 50%.

Una delle biometrie più utilizzate è il volto; quando si analizza il genere dal volto si opera con una biometria fisica.  
La struttura ossea di individui di genere maschile o femminile risulta essere differente e bisogna, tuttavia, tener conto che spesso il volto può essere alterato con tratti tipici della popolazione del sesso opposto a quello del soggetto (make-up, barba ecc...) *(Figura 1)*.

Figura

Immagine che contiene persona, guardando, indossando, faccia

Descrizione generata automaticamenteUna tecnica che potrebbe ovviare a problemi di «texture» che spesso si riscontrano in algoritmi di *gender recognition,* può essere l’utilizzo di landmark. I landmark sono infatti punti «chiave» del volto, generalmente 68, e dopo il loro rilevamento permettono di tralasciare dettagli del volto che in alcuni casi potrebbero essere fuorvianti.

Per studiare la distribuzione dei landmark sui volti si può far riferimento alla codifica di un algoritmo di *pose estimation.* L’algoritmo in questione utilizza un modello a ragnatela per suddividere il volto in diversi settori, all’interno dei quali ricadono i landmark.  
Il numero di settori della ragnatela è fissa per ogni volto, indipendentemente dalla dimensione del volto; il settore di appartenenza di ogni landmark viene stabilito mediante semplici formule relative alle coordinate cartesiane e alle circonferenze. In *figura 2* abbiamo un esempio in cui possiamo notare in verde i 68 landmark del viso e in rosso la ragnatela.

Figura

La ragnatela così ottenuta viene «srotolata» divenendo un vettore di interi i cui valori rappresentano il numero di landmark presenti in quel settore. Tale rappresentazione è molto compatta: se si vuole considerare come esempio una ragnatela composta da 4 fette per quadrante e 4 cerchi si avrà un array di 64 elementi.

Questo algoritmo ha inoltre prodotto ottimi risultati nella stima della posa di un soggetto; lo scopo di tale progetto è di testarne l’efficacia applbicandolo ad un problema di *gender classification.*

## Obiettivi

Al fine di ottenere la classificazione di genere si utilizza il dataset di immagini UTKFace. Tale obiettivo può essere raggiunto mediante utilizzo dei soli dati provenienti dall’algoritmo che applica la ragnatela ai volti presenti nell’immagine. Questo processo si divide in:

* estrazione del volto dall’immagine;
* stima della posizione dei landmarks del volto;
* creazione dell’array che rappresenta la ragnatela mediante l’algoritmo fornito;
* allenare un classificatore sugli array derivanti dall’algoritmo per la creazione della ragnatela;
* stimare l’accuratezza raggiunta per la *gender classification*.

Tale progetto ha coinvolto 2 candidati, pertanto è stato richiesto di:

* ottenere il dataset assegnato;
* estrarre dal dataset i volti dei soggetti;
* individuare la posizione dei landmark;
* utilizzare **più configurazioni** della ragnatela per creare diversi training set, validation  
   set e test set;
* creare **due classificatori** binari per determinare il genere, testandolo con i vari set sopra descritti;
* valutare le accuratezze relative alle differenti combinazioni e valutarne i pro e i contro.

## Dataset utilizzato

Il dataset UTKFace è un set di dati su volti in larga scala con intervallo di età tra 0 e 116 anni. Il dataset comprende oltre 20.000 immagini di volti etichettati con età, sesso ed etnia. Le immagini coprono grandi variazioni di posa, espressione facciale, illuminazione, risoluzione, ecc.

Le etichette riferite al volto contenuto in ogni immagine sono incluse nel nome del file e seguono tutte lo stesso schema, ovvero:

[età]\_[sesso]\_[razza]\_[data&ora].jpg

* [età] è un numero intero compreso tra 0 e 116 e indica l'età
* [sesso] è 0 (maschio) o 1 (femmina)
* [razza] è un numero intero compreso tra 0 e 4, che indica bianco, nero, asiatico, indiano e altri (come ispanico, latino, mediorientale).
* [data&ora] è nel formato di yyyymmddHHMMSSFFF, che mostra la data e l'ora in cui un'immagine è stata inserita all’interno del dataset.

Il dataset è fornito in due versioni differenti; la prima definita In-the-wild ovvero contenete l’immagine in formato integrale che comprende anche l’eventuale presenza di altri oggetti ed una seconda versione in cui le immagini contengono solo il volto allineato e ritagliato, tale procedimento è stato effettuato con Dlib.

Per perseguire lo scopo di tale progetto è stato utilizzata la seconda versione non avendo a disposizione ingenti risorse di calcolo.

Durante la fase di pre-processing è stata analizzata la distribuzione del dataset in base alle features indicate in etichetta per valutare quali potessero essere utili agli obiettivi prefissati; nella pagina seguente sono riportati grafici di tale analisi.

|  |
| --- |
|  |
| *Grafico 1* |
|  |
| *Grafico 2* |
|  |
| *Grafico 3* |

## Configurazioni assegnate

Le quattro configurazioni assegnate sono quelle con MAE (Minor Absolute Error) minore per la *pose estimation*, e sono le seguenti:

1. 4C\_4S\_var2
2. 4C\_4S\_var4
3. 4C\_3S\_inv
4. 5C\_4S\_inv

Il parametro C indica il numero di cerchi di cui è composta la ragnatela e il parametro S il numero di fette per ogni quadrante.  
Per quanto concerne la differenza tra la prima e la seconda configurazione, vi è una variazione del raggio dal centro della ragnatela ai cerchi.

Nella tabella 1 è possibile osservare la divisione del raggio per ogni configurazione:

|  |  |  |
| --- | --- | --- |
| **Configurazione** | **Divisione Raggio (dall’interno verso l’esterno)** | **Dimensione array** |
| *4C\_4S\_var2* | 8/15\*R; 12/15\*R; 14/15\*R; R | 64 |
| *4C\_4S\_var4* | 4/10\*R; 7/10\*R; 9/10\*R; R | 64 |
| *4C\_3S\_inv* | R/4; R/2; 3/4\*R; R | 48 |
| *5C\_4S\_inv* | R/5; 2/5\*R; 3/5\*R; 4/5\*R; R | 80 |

Tabella 1

# Preprocessing

## Algoritmo Ragnatela

All’inizio del progetto è stato fornito un algoritmo basato sulla configurazione 4 cerchi 4 fette per estrarre un “array ragnatela” a partire da un’immagine.

L'algoritmo è diviso in:

* face detection dell’immagine;
* estrazione dei 68 landmark del volto;
* assegnazione di ogni landmark al settore della ragnatela corrispondente.

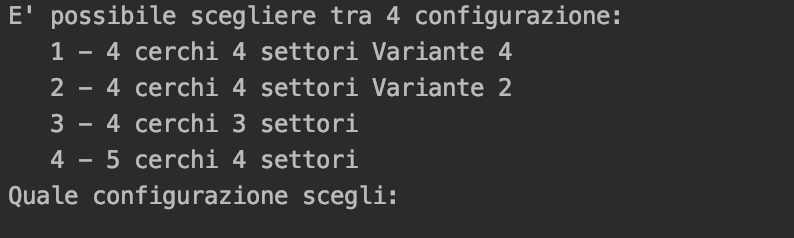
L'algoritmo assegnato si è rivelato essere una buona base di partenza per la costruzione di una soluzione responsive dell’algoritmo ragnatela, data l’assegnazione per l’utilizzo con diverse configurazioni della ragnatela, tuttavia, sono state necessarie diverse modifiche.

## Algoritmo Ragnatela 2.0

Al sopracitato algoritmo sono state aggiunte diverse funzionalità per renderlo più fruibile ed efficiente al raggiungimento degli obiettivi assegnati.

**Configurazioni Responsive**

La prima modifica apportata è stata l’aggiunta della funzione **scelta ()**, che permette di scegliere da linea di comando quale configurazione della ragnatela applicare al dataset. (*Figura 3*).



Figura

Per rendere totalmente responsive l’algoritmo è stato opportuno modificare in parte anche la funzione preesistente **aggiungi().**

**Responsive Resize**

Nell’algoritmo originale viene effettuato il resize delle immagini ad un numero fisso, pari 512 pixel.   
Nella versione di seguito illustrata, tale operazione è stata resa responsive, poiché con numero fisso non è stato possibile individuare i landmark su tutte le immagini. Inoltre, si è notato che abbassando la definizione a 256 pixel l’algoritmo è più veloce e non perde molta accuratezza.   
Per usufruire di tutte le immagini “utilizzabili” si è scelto di aumentare gradualmente la risoluzione delle immagini ogni qualvolta non era possibile, ad una data risoluzione partendo da 256 pixel, riconoscere un volto.   
Dopo vari test è emerso che le immagini che sono impossibili da processare ad una definizione di 2048 pixel, non sono utilizzabili, per tale motivo si è deciso di porre come limite proprio 2048 pixel.

In definitiva la funzione **responsive\_resize()** parte da un resize a 256 pixel e aumenta fino a 2048 pixel, passando per 512 e 1024 pixel.

**Analisi immagini non utilizzabili**

Anche applicando la funzione **responsive\_resize()** si è riscontrata una certa percentuale di immagini scartate dall’algoritmo. Approfondendo l’analisi, sono state riscontrate impurità nel dataset, come ad esempio volti parzialmente coperti, immagini con solo occhi e oggetti non inerenti a volti. Un’altra tipologia di impurità riscontrata è stata un’errata etichettatura di diverse immagini, infatti, per alcuni elementi mancano etichette nel nome file.  
Tali impurità sono individuate durante la fase di recupero delle etichette e non vengono aggiunte al dataset.

Di seguito sono riportati alcuni esempi di impurità:

|  |  |  |
| --- | --- | --- |
| Immagine che contiene interni, guardando, vicino, primopiano  Descrizione generata automaticamente | Immagine che contiene persona, indossando, vicino, faccia  Descrizione generata automaticamente | Immagine che contiene uccello, cibo, sfocato, rosso  Descrizione generata automaticamente |
| Immagine che contiene persona, uomo, guardando, indossando  Descrizione generata automaticamente | **Immagine che contiene metro  Descrizione generata automaticamente** | Immagine che contiene sedendo, vicino, remoto, tavolo  Descrizione generata automaticamente |

**Salvataggio Dati**

Infine, è stata aggiunta la funzione **write\_list\_to\_file()** che salva le informazioni date in output dall’algoritmo in un file formato .csv.

# Classificatori

In tale capitolo si analizzerà in dettaglio il percorso di ricerche effettuate per raggiungere la soluzione migliore in termini di accuratezza.

Le tipologie di modelli di apprendimento supervisionati sviluppati per la risoluzione del problema di classificazione sono:

* Support Vector Machine (SVM);
* rete neurale con vettore ragnatela;
* rete neurale convoluzionale;
* rete neurale ibrida immagine e ragnatela.

Nei successivi paragrafi si procederà ad analizzare nel dettaglio ogni tipologia di modello ed i relativi risultati utilizzando le configurazioni assegnate.

## Classificatore SVM

Le **Support Vector Machines** (**SVM**) sono dei modelli di apprendimento supervisionato associati ad algoritmi di apprendimento per la classificazione e la regressione. Dato un insieme di esempi per l'addestramento, ognuno dei quali etichettato con la classe di appartenenza fra le due possibili classi, un algoritmo di addestramento per le SVM costruisce un modello che assegna i nuovi esempi a una delle due classi, ottenendo quindi un classificatore lineare binario.   
Un modello SVM è una rappresentazione degli esempi come punti nello spazio, mappati in modo tale che gli esempi appartenenti alle due diverse categorie siano chiaramente separati da uno spazio il più possibile ampio. I nuovi esempi sono quindi mappati nello stesso spazio e la predizione della categoria alla quale appartengono viene fatta sulla base del lato nel quale ricade.

Nel nostro progetto l’SVM è stata la prima soluzione implementata e testata. Non ha prodotto risultati ottimi, considerando il punteggio di accuratezza pari a 0.74.

Di seguito i risultati ottenuti testando le 4 configurazioni:

|  |  |
| --- | --- |
| **Configurazione** | **Accuracy** |
| 4C\_4S\_var4 | 0.72 |
| 4C\_4S\_var2 | 0.73 |
| 5C\_4S\_inv | 0.74 |
| 4C\_3S\_inv | 0.71 |

Tabella

## Reti neurali

Le reti neurali artificiali sono modelli matematici composti da neuroni artificiali di ispirazione alle reti neurali biologiche e sono utilizzate per risolvere problemi ingegneristici di Intelligenza Artificiale.

Volendo dare una definizione più dettagliata potremmo dire che **le reti neurali sono modelli di calcolo matematico-informatici basati sul funzionamento delle reti neurali biologiche, ossia modelli costituiti da interconnessioni di informazioni**; tali interconnessioni derivano da neuroni artificiali e processi di calcolo basati sul modello delle scienze cognitive chiamato “connessionismo”.

## Rete neurale con vettore ragnatela

Il secondo modello sviluppato utilizza una rete neurale a cui viene dato in input il dataset ricevuto in output dall’algoritmo “ragnatela”, quindi, per ogni immagine abbiamo in input un array di interi con un size variabile in base alla configurazione scelta *(Tabella 1)*.

Per massimizzare l’accuratezza dei risultati sono stati variati in diversi modi i seguenti parametri:

* batch size ed Epoch del Fit;

percentuale di dropout ad ogni layer;

* batch normalization dei layer intermedi;
* percentuale di elementi dati in Validation e in Test;
* dimensione e numero dei layer intermedi;
* tipologia di optimizer per la compilazione del classificatore;
* tipologia di loss per la compilazione del classificatore;
* tipologia di funzione di attivazione.

È stata inoltre tentata l’aggiunta delle due features riguardanti età e razza dei volti all’array ragnatela, senza notare rilevanti miglioramenti.

I risultati ottenuti sono indicati in tabella 3:

|  |  |  |
| --- | --- | --- |
| **Configurazione** | **Loss** | **Accuracy** |
| 4C\_4S\_var4 | 0.54 | 0.72 |
| 4C\_4S\_var2 | 0.54 | 0.73 |
| 5C\_4S\_inv | 0.50 | 0.75 |
| 4C\_3S\_inv | 0.55 | 0.70 |

Tabella

Possiamo quindi affermare che anche con l’utilizzo di reti neurali non è stato possibile ottenere una percentuale di accuratezza maggiore dello 0.75.

## Rete neurale convoluzionale

Per conoscere la differenza di accuratezza tra una rete neurale basata sul solo array ragnatela e una rete neurale che prende in input l’intera immagine è stata realizzata una rete neurale convoluzionale per la classificazione dei generi.  
La **Convolutional Neural Network** (**CNN**) rappresenta un’architettura di rete neurale artificiale di grande successo nelle applicazioni di visione artificiale e ampiamente utilizzata anche in applicazioni che processano media come audio e video.  
L’applicazione più popolare di rete neurale convoluzionale resta comunque quella di *identificare,* da parte di un computer, cosa un’immagine rappresenta.  
  
Per massimizzare l’accuratezza del risultato sono stati variati i seguenti parametri:

* batch size ed Epoch del Fit;
* percentuale di dropout;
* batch normalization dei Layer intermedi;
* percentuale di elementi dati in Validation ed in Test;
* dimensione e numero dei layer intermedi;
* tipo di funzione di attivazione.

L’accuracy riscontrata utilizzando questa tipologia di rete neurale è 0.88.

## Rete neurale ibrida

Un passo successivo è stata l’implementazione di una rete neurale ibrida, che unisce la rete neurale convoluzionale, che prende in input le sole immagini, con la rete neurale che prende in input l’array ragnatela. Le due reti si concatenano in un'unica rete che fornirà in output un risultato unico.

Il modello della rete neurale convoluzionale utilizzato è il medesimo della rete convoluzionale analizzata nella soluzione precedente, in modo da poter riscontrare miglioramenti dovuti alla sola aggiunta della rete neurale con input l’array ragnatela.

Anche in questo caso per massimizzare l’accuratezza del risultato sono stati variati i seguenti parametri:

* patch e numero di epoche del Fit;
* percentuale di dropout;
* batch normalization dei layer intermedi;
* percentuale di elementi dati in Validation e in Test;
* dimensione e numero dei layer intermedi;
* tipo di funzione di attivazione.

I risultati ottenuti sono indicati in tabella:

|  |  |  |
| --- | --- | --- |
| **Configurazione** | **Loss** | **Accuracy** |
| 4C\_4S\_var4 | 0.25 | 0.89 |
| 4C\_4S\_var2 | 0.27 | 0.88 |
| 5C\_4S\_inv | 0.27 | 0.88 |
| 4C\_3S\_inv | 0.26 | 0.88 |

Tabella

# Ulteriori Evidenze (dataset CelebA)

Nel tentativo di ottenere una percentuale migliore in termini di accuratezza si è deciso di sottoporre in input al modello, con configurazione migliore nel test sulla rete neurale con sola ragnatela, una mole maggiore di dati per l’addestramento. Per questo motivo è stato scelto l’utilizzo del dataset CelebA.

## Dataset CelebA

CelebFaces Attributes Dataset (CelebA) è un set di dati di volti con attributi su larga scala con oltre 200.000 immagini di celebrità, ognuna con 40 annotazioni di attributi. Le immagini in questo dataset coprono grandi variazioni di posa e di sfondi.

Tale dataset differisce da UTKFace per la tipologia di attributi utilizzati per etichettare le immagini e dalla dimensione dello stesso dataset; in UTKFace le etichette riguardano riferimenti biologici (razza, età e sesso), mentre in CelebA abbiamo un’attenzione maggiore sugli attributi riguardanti l’aspetto conferito alla persona da oggetti o segni di riconoscimento particolari (capelli ricci, occhiali, calvizie, baffi, make-up, etc..).

## “Re-allenamento” modello

Avendo prima preprocessato il dataset con l’algoritmo ragnatela per ottenere l’array per ogni immagine, e in seguito allenato nuovamente il classificatore con i nuovi dati si è giunti al seguente risultato:

|  |  |  |
| --- | --- | --- |
| **Configurazione** | **Loss** | **Accuracy** |
| 5C\_4S\_inv | 0.42 | 0.81 |

Tabella

## Test Incrociati

Infine, si è provveduto a testare il classificatore allenato sul dataset UTKFace sul dataset CelebA e viceversa. Le evidenze che emergono da tali test sono le seguenti:

|  |  |  |
| --- | --- | --- |
| **Classificatore allenato su** | **Dataset di test** | **Accuracy** |
| UTKFace | UTKFace | 0.74 |
| CelebA | CelebA | 0.81 |
| UTKFace | CelebA | 0.74 |
| CelebA | UTKFace | 0.73 |

Tabella

Dalla tabella 6 è emerso che il classificatore addestrato con UTKFace raggiunge un punteggio di accuratezza che resta invariato su entrambi i dataset di test.

Invece, nonostante il classificatore addestrato con CelebA abbia raggiunto un’accuratezza di 0.81 in fase di test con il proprio dataset, con il test incrociato tale valore diminuisce. Tale diminuzione è imputabile alla differenza di composizione dei due dataset; in UTKFace sono presenti volti in una fascia di età che spazia tra gli 0 e i 116 anni, con una distribuzione per razza diversificata. In CelebA, invece, abbiamo un range di età più ristretto, in cui non sono inclusi volti in età infantile ed avanzata.

# Conclusioni

La ricerca della soluzione migliore è iniziata con l’utilizzo di classificatori di tipo SVM; tale tipologia ha dimostrato, sulle 4 configurazioni utilizzate, un’accuratezza massima di 0.74.

Ritenendo non ottimale il risultato raggiunto, si è proseguito con l’implementazione di un classificatore basato su una rete neurale che prende in input il dataset dato in output dall’ algoritmo “ragnatela”. Anche in questo caso, testando l’algoritmo sulle 4 configurazioni, non si è superato il limite di accuratezza di 0.75.

Per conoscere la differenza di accuratezza tra una rete neurale basata sul solo array ragnatela e una rete neurale che prende in input l’intera immagine è stata realizzata una rete neurale convoluzionale per la classificazione dei generi che ha portato ad un’accuratezza di 0.88.

Successivamente è stata l’implementata una rete neurale ibrida, che unisce la rete neurale convoluzionale con la rete neurale che prende in input l’array ragnatela; tale modello produce un’accuratezza pari a 0.89. Seppur piccola, la variazione di accuratezza tra i due ultimi modelli dimostra un leggero contributo derivante dall’utilizzo della ragnatela.

Nel tentativo di ottenere una percentuale migliore in termini di accuratezza nel modello della rete neurale con solo vettore ragnatela, si è deciso di sottoporre in input a tale modello una mole maggiore di dati con l’utilizzo del dataset CelebA.

Questo tentativo ha prodotto un'accuratezza di 0.81, dato che risulta, però, inficiato dalla presenza all’interno di CelebA di volti raffiguranti solo persone in un range di età più stretto rispetto ad UTKFace.

# Appendice

## Grafici della Rete Neurale con Vettore Ragnatela - UTKFace

**Configurazione 4C\_4S\_var4**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene mappa, screenshot  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

**Configurazione 4C\_4S\_var2**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene testo, mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

**Configurazione 5C\_4S\_inv**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene mappa  Descrizione generata automaticamente** | **Immagine che contiene mappa  Descrizione generata automaticamente** |

**Configurazione 4C\_3S\_inv**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene testo, mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

## Grafici della Rete Neurale Convoluzionale – solo immagini

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene testo, mappa  Descrizione generata automaticamente** | **Immagine che contiene testo  Descrizione generata automaticamente** |

## Grafici della Rete Neurale Ibrida

**Configurazione 4C\_4S\_var4**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

**Configurazione 4C\_4S\_var2**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene testo, mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

**Configurazione 5C\_4S\_inv**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

**Configurazione 4C\_3S\_inv**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
| **Immagine che contiene mappa  Descrizione generata automaticamente** | **Immagine che contiene testo, mappa  Descrizione generata automaticamente** |

## Grafici della Rete Neurale con Vettore Ragnatela – Dataset CelebA

**Configurazione 5C\_4S\_inv**

|  |  |
| --- | --- |
| *Accuratezza* | *Loss* |
|  |  |