1. Ricerca del miglior numero di nodi per i layers convoluzionali

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # nodi | Dataset | Epoche | TrainAcc/loss | ValAcc/loss | TestAcc | TestLoss | Durata |
| 32 | 10Rand | 10 | 0.9926/0.0264 | 0.9965/0.0129 | 0.9938 | 0.0211 | 01:06 |
| 32 | 10Poor | 10 | 0.9875/0.0552 | 0.9858/0.0696 | 0.9844 | 0.0637 | 00:16 |
| 32 | 43Class | 10 | 0.9875/0.0432 | 0.9938/0.0275 | 0.9837 | 0.0653 | 02:10 |
| 64 | 10Rand | 10 | 0.9949/0.0168 | 0.9974/0.0068 | 0.9931 | 0.0198 | 01:42 |
| 64 | 10Poor | 10 | 0.9962/0.0242 | 0.9559/0.1282 | 0.9518 | 0.1697 | 00:20 |
| 64 | 43Class | 10 | 0.9933/0.0232 | 0.9924/0.0240 | 0.9842 | 0.0619 | 03:27 |
| 128 | 10Rand | 10 | 0.9962/0.0125 | 0.9891/0.0350 | 0.9800 | 0.0737 | 03:02 |
| 128 | 10Poor | 10 | 0.9890/0.0458 | 1.0000/0.0196 | 0.9943 | 0.0263 | 00:33 |
| 128 | 43Class | 10 | 0.9943/0.0187 | 0.9880/0.0390 | 0.9737 | 0.110 | 06:15 |

Osservando le accuracy di test per i vari dataset a parità di nodi nei livelli convoluzionali, risulta evidente come la rete con 32 nodi per livello sia la più efficiente, coniugando una velocità prestazionale maggiore rispetto alle altre due e raggiungendo ugualmente precisione e loss molto soddisfacenti.

1. Ricerca del miglior numero di nodi per il penultimo layer fully-connected

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # nodi | Dataset | Epoche | Accuracy | Loss | Durata |
| 1024 | 10Random | 10 | 0.9936 | 0.0182 | 01:10 |
| 512 | 10Random | 10 | 0.9935 | 0.0195 | 01:05 |
| 256 | 10Random | 10 | 0.9928 | 0.0226 | 01:03 |

A parità di dataset e di epoche, i dati mostrano come sia preferibile un penultimo layer Dense costituito da 512 nodi. Le performance del modello con tale caratteristica raggiungono quelle in cui è adottato un livello da 1024 nodi, inoltre è presente un risparmio computazionale e temporale lieve ma comunque significativo e vantaggioso per accelerare l’esecuzione dei futuri test da effettuare. Pertanto, la ricerca del miglior numero di nodi per il penultimo fully-connected ha evidenziato un ottimo compromesso tra precisione e velocità con 512 nodi.

1. Analisi dell’early stopping

Pertanto, a partire dalla CNN con layer convoluzionali formati da 32 nodi, è stato testato l’early stopping per analizzarne i risultati. Il numero di epoche è aumentato a 15.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Epoche | EarlyStop | TrainAcc/loss | ValAcc/loss | TestAcc | TestLoss | Durata |
| 10Random | 15 | No | 0.9924/0.0261 | 0.9961/0.0111 | 0.9916 | 0.0270 | 01:35 |
| 10Random | 15 | Si (11) | 0.9916/0.0269 | 0.9862/0.0511 | 0.9956 | 0.0196 | 01:17 |
| 10Poorest | 15 | No | 0.9970/0.0180 | 1.0000/0.0151 | 0.9958 | 0.0023 | 00:20 |
| 10Poorest | 15 | Si (14) | 0.9978/0.0139 | 0.9969/0.0423 | 0.9844 | 0.0564 | 00:19 |
| 43Classes | 15 | No | 0.9939/0.0206 | 0.9947/0.0207 | 0.9857 | 0.0582 | 03:14 |
| 43Classes | 15 | Si (12) | 0.9892/0.0348 | 0.9892/0.0324 | 0.9800 | 0.0816 | 02:40 |

Dai risultati riportati è evidente come una rete di così piccole dimensioni non sfrutti tutti i benefici che normalmente sono introdotti con l’early stopping. La precisione acquisita non varia di molto con il suo utilizzo; tuttavia, si è ritenuto appropriato adottare tale tecnica come forma di prevenzione dall’overfitting e per ridurre la durata degli esperimenti con il relativo costo computazionale.

1. Confronto tra i risultati delle funzioni di attivazione “ReLU” e “LeakyReLU”

Un terzo test effettuato ha coinvolto le funzioni di attivazione dei tre livelli convoluzionali e del penultimo Dense che compongono la rete. Finora è sempre stata utilizzata la ReLU (Rectified Linear Unit). È stata ora testata invece la LeakyReLU e ne sono riportati i risultati nella seguente tabella.



Nonostante la funzione LeakyReLU sia leggermente più onerosa, l’early stopping ha permesso una riduzione della durata temporale dell’esperimento, dal momento che raggiunge l’ottimo prima rispetto ad una rete con funzione di attivazione ReLU. Tuttavia, i risultati conseguiti non sembrano differenziarsi troppo da quelli effettuati nei test precedenti. Di conseguenza, si è scelto di adottare entrambe le funzioni per poter effettuare una scelta discriminante nel seguito dell’esperimento.

Potendo riassumere i test svolti finora, una configurazione ottimale della semplice rete neurale convoluzionale sembra essere quella con livelli formati da 32 nodi e che usa il metodo dell’early stopping.

1. //direttamente nella relazione

//da qui in poi parte quella in comune con MobileNetV2

L’analisi ora verte sull’utilizzo o meno della augmentation nei generatori di train e test del dataset utilizzato. Una buona augmentation permette alla rete di osservare cartelli stradali in molteplici forme, simulandone una percezione durante la guida su strada. I test effettuati, portando avanti anche l’interscambio delle funzioni di attivazione ReLU e LeakyReLU, hanno registrato i seguenti risultati:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Act Fn | TrainAug | TestAug | TrainAcc/loss | ValAcc/loss | TestAcc | TestLoss | Durata |
| 10Rand | ReLU | Si | No | 0.9784/0.0690 | 0.9552/0.1394 | 0.9860 | 0.0453 | 01:14 |
| 10Rand | Leaky | Si | No | 0.9865/0.0422 | 0.9609/0.1417 | 0.9919 | 0.0346 | 01:50 |
| 10Rand | ReLU | No | Si | 0.9911/0.0349 | 0.9816/0.0657 | 0.7722 | 1.4146 | 00:43 |
| 10Rand | Leaky | No | Si | 0.9931/0.0243 | 0.9849/0.0444 | 0.8115 | 1.1947 | 00:48 |
| 10Rand | ReLU | Si | Si | 0.9832/0.0533 | 0.9740/0.1036 | 0.9869 | 0.0591 | 01:39 |
| 10Rand | Leaky | Si | Si | 0.9625/0.1154 | 0.9425/0.1879 | 0.9754 | 0.0790 | 00:48 |
| 10Poor | ReLU | Si | No | 0.9547/0.1463 | 0.9339/0.2291 | 0.9788 | 0.1204 | 00:14 |
| 10Poor | Leaky | Si | No | 0.9596/0.1391 | 0.9008/0.3408 | 0.9632 | 0.1172 | 00:20 |
| 10Poor | ReLU | No | Si | 0.9975/0.0164 | 0.9969/0.0320 | 0.7691 | 1.3339 | 00:14 |
| 10Poor | Leaky | No | Si | 0.9821/0.0721 | 0.9717/0.0924 | 0.7408 | 1.1679 | 00:10 |
| 10Poor | ReLU | Si | Si | 0.9354/0.1974 | 0.8866/0.3076 | 0.9419 | 0.1935 | 00:14 |
| 10Poor | Leaky | Si | Si | 0.9712/0.1106 | 0.9386/0.2357 | 0.9773 | 0.1027 | 00:16 |
| 43Classes | ReLU | Si | No | 0.9674/0.0999 | 0.9539/0.1733 | 0.9730 | 0.0997 | 02:54 |
| 43Classes | Leaky | Si | No | 0.9761/0.0762 | 0.9563/0.1691 | 0.9761 | 0.0904 | 03:19 |
| 43Classes | ReLU | No | Si | 0.9903/0.0321 | 0.9900/0.0329 | 0.6926 | 2.4078 | 02:00 |
| 43Classes | Leaky | No | Si | 0.9896/0.0361 | 0.9820/0.0569 | 0.6860 | 2.1190 | 01:23 |
| 43Classes | ReLU | Si | Si | 0.9373/0.2003 | 0.9147/0.2855 | 0.9430 | 0.1952 | 01:38 |
| 43Classes | Leaky | Si | Si | 0.9727/0.0886 | 0.9442/0.2162 | 0.9579 | 0.1608 | 02:51 |

I valori qui riportati mostrano, come prevedibile, un lieve decremento prestazionale dovuto all’augmentation: cartelli stradali modificati sono più difficili da riconoscere. Tuttavia, si ritiene necessario effettuare tale fase di pre-processing in quanto è estremamente utile per un utilizzo pratico della rete su strada, rendendola in grado di raggiungere ottime prestazioni anche in applicazioni reali. Se così non fosse, ci si ricondurrebbe al caso in cui la augmentation è effettuata sul solo test set, casistica in cui appare immediato come la rete, non osservando mai in fase di addestramento i cartelli che più rispecchiano situazioni reali su strada, raggiunga mediocri risultati (conclusioni ovviamente da evitare in caso di utilizzo pratico). Inoltre, il test ha mostrato come una funzione di attivazione LeakyReLU, su cui prima vi era indecisione, sia preferibile ad una normale ReLU, raggiungendo, seppur di poco, risultati più soddisfacenti.

Concludendo i test casistici che hanno coinvolto la PersonalCNN, la configurazione ottima di tale rete risulta essere dunque quella con:

* 32 nodi nei livelli convoluzionali
* Early stopping
* Funzione di attivazione LeakyReLU