Progetto 6
Attacchi avversari alla classificazione di immagini

F. Iannaccone, M. Conti

Obiettivo di questo progetto è attaccare una rete neurale convoluzionale per il riconoscimento di volti di animali. In questo progetto si considera il dataset LHI-Animal-Faces proposto in [1]. Il dataset è costituito da circa 2200 immagini con 19 categorie di animali e una di volti umani, come mostrato in figura 1. Per la classificazione utilizzerete l'architettura mostrata in tabella 1.

In questo progetto i passi da seguire sono:

1. **Download dei dati**. Il dataset può essere scaricato al seguente link:
   http://www.stat.ucla.edu/ jxie/iFRAME/multiclassClassification/animalFaceClassification.html

2. **Preparazione dei dati**. Il dataset va suddiviso considerando i 4/6 di immagini per il training, 1/6 per la validation e 1/6 per il test. Preparate i dati per le 20 classi utilizzando `ImageDataGenerator` con il metodo `flow_from_directory`. Per tutti i set, riportate le immagini nel range $[0, 1]$ e ridimensionate le immagini a 256×256 pixel. Solo per il set di training, prevedete le seguenti operazioni di data-augmentation: rotazione random da -5 gradi a 5 gradi, ridimensionamento con fattore random nel range [0.8, 1.2] e riflessione orizzontale random.

3. **Architettura**. Definite l'architettura descritta in tabella 1.

4. **Addestramento**. Per l'addestramento utilizzate l'ottimizzatore Adam tramite la funzione di Keras `keras.optimizers.Adam`. Mentre per la loss function adottate la Cross-Entropy. Utilizzate le prestazioni sul set di validazione per selezionare i migliori valori per il learning-rate, il batch-size, il numero di epoche.

5. **Valutazione delle prestazioni**. Utilizzate il test-set per valutare le prestazioni in termini di accuratezza. Inoltre calcolate la matrice di confusione tramite il metodo:
   `sklearn.metrics.confusion_matrix`

6. **Attacco**. Usate il toolbox Foolbox per attaccare con la tecnica FGSM (Fast Gradient Signed Method) [2] le immagini del test-set e valutate le prestazioni dopo l'attacco.

7. **Attacco Target**. Con il toolbox Foolbox è possibile anche eseguire attacchi target, dove le immagini vengono modificate per essere predette appartenenti ad un classe target. As esempio, un'immagine della classe gatto (classe di partenza) può essere appositamente modificata per essere predetta dalla CNN come cane (classe target). Per eseguire l'attacco target in Foolbox dovete fornire alla funzione di attacco il risultato della seguente istruzione:

Figure 1: Le categorie del dataset LHI-Animal-Faces.

Tipo	Dim. Spaziale	Num. Feat.	Attivazione	Dim. Uscita
Convolution	5×5	128	ReLU	(256, 256, 128)
Max-pooling	2×2	-	-	(128, 128, 128)
Convolution	5×5	128	ReLU	(128, 128, 128)
Max-pooling	2×2	-	-	(64, 64, 128)
Convolution	5×5	256	ReLU	(64, 64, 256)
Max-pooling	2×2	-	-	(32, 32, 256)
Convolution	5×5	256	ReLU	(32, 32, 256)
Max-pooling	2×2	-	-	(16, 16, 256)
Convolution	3×3	512	ReLU	(16, 16, 512)
Fully-connected	-	2048	ReLU	(2048)
Fully-connected	-	512	ReLU	(512)
Fully-connected	-	20	Softmax	(20)

Table 1: Architettura della CNN.

```
criteria = foolbox.criteria.TargetedMisclassification(convert_to_tensor(y_target))
```

dove `y_target` contiene le classi target per le immagini da attaccare. Provate l'attacco target fra 5 coppie di classi a vostra scelta.

References

[1] Z. Si, and S.-C. Zhu, "Learning Hybrid Image Templates (HIT) by Information Projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012

[2] I.J. Goodfellow and J. Shlens and C. Szegedy, "Explaining and Harnessing Adversarial Examples," International Conference on Learning Representations (ICLR), 2015