Snakes' species classification

Grazia Noemi Compagnino noemicompagnino97@live.it
Maqbool Thoufeeq Tharayil thrmbl95b15z222z@studium.unict.it

1. Snakes' species model description

We describe here **SnakeImages**, a dataset that consists in RGB images of snakes, divided in 5 classes on the basis of their corresponding species and geographic location (continent, country). The goal is to create a system capable to automatically categorize snakes on the species class (see Fig.1). This analysis has the scope to explore how can be reduced erroneous and delayed healthcare actions and improved snakebite eco-epidemiological data thanks to the help of Machine Learning's snake identification. Initially, input data flows trough a 2D encoder, to exploit inter slice volumetric information. Then the extracted features are provided to a capsules-based encoder, which predicts treatment outcome as positive or negative. It presents an initial 2D encoder of five convolutional layers, to process the input

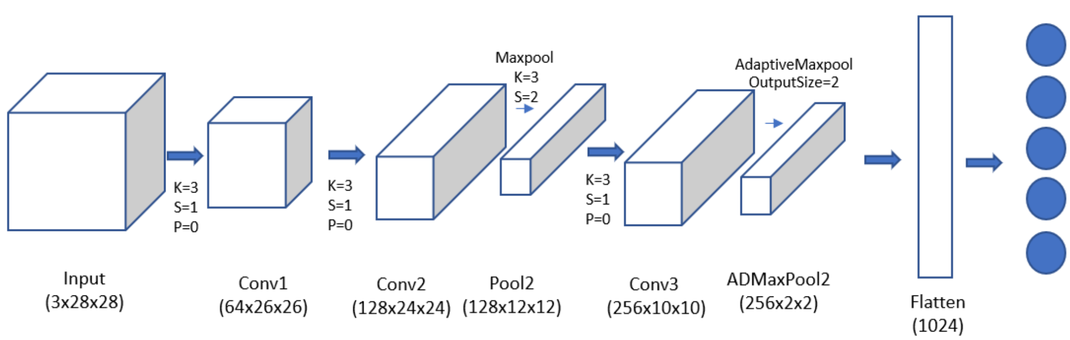

Figure 1: A simple representation of the proposed model.

volume. The five 2D convolution layers have respectively 3, 64, 128, 256 and 512 channels, 3x3x3x3x1 (we have 81 parameters) kernels applied with a stride of 1 in each dimension and followed by a ReLU activation function.

2. Snakes' species dataset

The majority of the data were gathered from online biodiversity platforms (i.e.,iNaturalist, HerpMapper). The **SnakeImages** dataset consists of 17389 snakes images, 384*384 pixels (a commonly used size for CNN inputs), containing the representation of a species of snake.

Each training class contains more than 3000 samples. We procedurally divided the dataset in training set, test set and validation set, for 0.8, 0.1 and 0.1 fraction respectively. So, we will have 13913 images in training sample, and 1738 in validation and test samples. The snake species we can see are:

- **Class 1:** Nerodia Sipedon - Northern Watersnake
- **Class 2:** Thamnophis Sirtalis - Common Garter snake
- **Class 3:** Storeria Dekayi - DeKay's Brown snake
- **Class 4:** Patherophis Obsoletus - Black Rat snake
- **Class 5:** Crotalus Atrox - Western Diamondback rattlesnake

So, when we import the dataset, we transform each image samples into 64x64 size RGB initially for sample representation. After that, when we add it to the train set, we transform the image sample to 28x28 to Tensor format which is normalized in 0.5x0.5.

Figure 2: A random sample of snake.

3. Training procedure

We employ data augmentation mechanisms like random input rotation to reduce overfitting issues. We use 2D Maxpooling technique for down sampling the input along its spatial dimensions of Height and Width by taking the maximum value over each input window for each channel of input.
Stochastic Gradient Descent (SGD) is chosen as optimizer algorithm using a learning rate of 0.01. Then we compute the Cross-Entropy Loss between input and target, useful to train a classification problem with C classes.

Also we use Adaptive Average Maxpooling which is an average pooling operation that, given an input and output dimensionality, calculates the correct kernel size necessary to produce an output of the given dimensionality from the given input. We create a fully connected layer by applying linear transformation with several input features and 5 output features with a ReLU activation function.

All the models are trained from scratch for 10 epochs on a Nvidia GeForce RTX 2060 with 32GB of VRAM.

4. Experimental Results

Several experiments conducted by us to test the effectiveness of the proposed **SnakeImages** model in comparison to traditional approaches. The baseline network and the **SnakeImages** were trained and tested individually. Models were trained as described in the previous section.

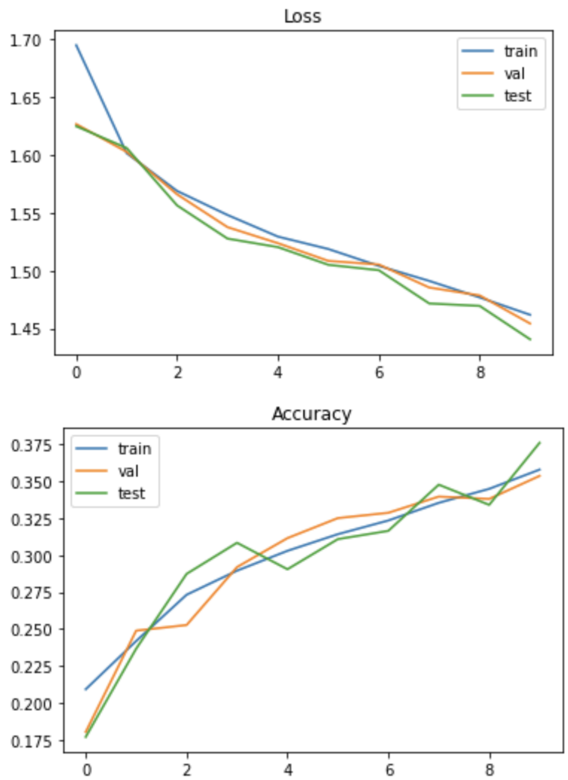

Figure 3: Best model.

We choose as the best model the one with 1 convolution layer, a Fully-Connected layer (FC) and the classifier. By the plot we can see that our model is predicted with an accuracy test of 34.91% and that the accuracy increase in accordance to the number of epochs. Table 1 shows test results for the proposed model as well as of ablation studies (i.e., different variants of the final model when adding or removing layers).

Model	Accuracy Test
Baseline Net	25.63%
– + Conv Layer 1 + FC + classifier	34.91%
– + Conv Layer 2 + FC + classifier	33.35%
– + Conv Layer 3 + FC + classifier	29.87%
– + Conv Layer 4 + FC + classifier	30.61%
– + Conv Layer 5 + FC + classifier	25.63%
Your final model	34.91%

Table 1: Test performance of models.