PowerAI Vision Workshop Exercise 1 - Image classification

In this exercise you will practice Image Classification. With Image Classification, a ML algorithm learns to assign categories (classes) to images. The goal of the exercise is to familiarize with the user interface, Image Classification related tasks and the overall process.

1. Creata a **dataset** called Sheep.
   a. On the welcome screen, press ’Get Started’ (alternatively, select Data Sets from the menu bar). The Data Sets screen appears.
   b. Press the plus sign to ’Create a new dataset’ on the leftmost tile. (.zip file upload is valid only for datasets exported from PAIV). Name the data set ’Sheep’.
   c. Click on the tile ’Sheep’. A ’Data set / Sheep’ screen will open.
   d. Click on ’Import files’ under the leftmost tile ’Drop files here’.
   e. On the leftmost pane, Click on Categories and select ’Single’. We will train our model with single sheep images.
   f. At the bottom of the pane, select ’Items per page: 100’ to view as many images as possible. This will likely improve productivity.
   g. Remove some selected images from the data set that you still keep on your machine. These may be used in final testing.
2. Create **classes** and **classify** images.
   a. On the left pane, expand Categories. Click on the button ’Add category’.
   b. Add two categories: ’blackhead’ and ’northern’.
   c. Select the images with Blackhead Persian sheep. Once selected all matching images, click ’Assign category’. Select blackhead and press ’Assign’. The image category bar changes to the new category name.
   d. Repeat step c. to assign the rest to the ’northern’ category.
   e. Select the category ’general’
   f. Select all images from the category ’general’ and delete them.
   g. Remove category labels ’general’ and ’single’.
3. **Augment** data
   a. Select the ’blackhead’ images

   b. Select All
   c. Click ’Augment Data’
   d. Select Noise and Horizontal Flip. Leave the parameters unchanged.
   e. Click Continue.
   f. Name the new dataset ’blackhead_augmented’.
   g. Once augmentation is complete, a new data set is created. This new dataset contains the original images **plus** the augmented ones.

4. **Train** your model
   a. In your dataset, press the ’Train’ button.
   b. Name the new model ’Blackheads_classifier’.
   c. Select Image Classification as the Type of Training.
   d. Set ’max iteration’ to a value between 600 and 800
   e. Leave other parameters unchanged.
      i. Note: Advanced parameters can improve but even inhibit training. Base models are pretrained models that can speed up the training process if there are similarities between the actual data set and the base model.
   f. Press ’Train’ to schedule training, and monitor the training process. After initialization the training starts and training loss is shown vs iteration.Training loss and accuracy improves, whereas Test Loss varies.
   g. Once Train Loss is eliminated, your model learned what can be learned from this data. Test Loss may not be very good. This can be a sign of overfitting (insufficient data). Wait until the model training completes.

5. **Evaluate** the model.
   a. Click ’Model details’
   b. Check the training results (evaluation metrics) and the confusion matrix (it is only visible if the Advanced metric option is set to ’On’).
   c. Observe the Categories and the included images. You see that the blackhead category is overrepresented (In our case due to augmentation). There is a slight chance that the classifier has become skewed. (Understands ’blackhead’ much better then ’norhtern’). It would have been a good idea to equally augment each category.

        i. Note: you can export the model to transfer it to another PAIV machine (e.g. for inference). The model is created as a password-protected zip file. The password is undisclosed, and the zip file cannot be used elsewhere.
        ii. Custom models based on Tensorflow can be exported. They can be deployed anywhere, not only on PAIV systems.

6. **Deploy** and **test** your classifier.
    a. Click 'Deploy model'. Assign a name to the deployed model.
    b. After the deployment finishes, click the name of your newly created model.
    c. Observe that an API endpoint is created that accepts your service request and ready to use the classifier for data sent.
    d. Upload an image from the input data.
    e. The model assigns a category, a confidence level and shows on the heat map which part of the image contributed most to the final decision.
        i. Note: this training, although accuracy could have reached a high value, is not a good predictor. It produces many errors and does not generalise well.

This concludes the exercise.