From running “python check\_images.py”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CNN Model Architecture** | **% Match**  **Labels** | **% Dogs**  **Correct** | **% Breeds**  **Correct** | **% Not-a-Dog**  **Correct** |
| Resnet | 12.5 | 100.0 | 16.7 | 32.1 |
| AlexNet | 12.5 | 100.0 | 16.7 | 35.7 |
| VGG | 15.0 | 100.0 | 16.7 | 35.7 |

The “best” model architecture is VGG. It classifies % match at 15% while the other two – ResNet and AlexNet classify it at 12.5%. While % Dogs Correct and % Breeds Correct register the same values for all three CNN model architectures, we see that VGG and AlexNet classify % Not-a-Dog Correct at 35.7% and ResNet classifies that at 32.1%. Also, I notice that for the same operation, VGG took 0:0:38 s, much faster than ResNet’s 0:0:8 s and AlexNet’s 0:0:4 s.

Clearly, the model VGG is the one that has outstanding performance in terms of timing and accuracy in classifying dogs, breed, match and not-a-dog.

From running “ sh run\_models\_batch\_uploaded.sh”:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CNN Model Architecture** | **% Match**  **Labels** | **% Dogs**  **Correct** | **% Breeds**  **Correct** | **% Not-a-Dog**  **Correct** |
| Resnet | 0.0 | 0.0 | 0.0 | 50.0 |
| AlexNet | 0.0 | 0.0 | 0.0 | 50.0 |
| VGG | 0.0 | 0.0 | 0.0 | 50.0 |

Resnet, AlexNet and VGG correctly identified my Dog\_01 and Dog\_02 (upside down image of Dog\_01) as maltese. AlexNet identified my animal, the Bald Eagle as a kite and Eiffel Tower as an apron. Resnet identified Bald Eagle as a kite and Eiffel Tower as a modem. VGG identified Bald Eagle as a kite and Eiffel Tower as a steel arch bridge. Across the board, the 3 CNN model architectures identify dog images well, but are not performing as well for animals and objects. Out of the 3, AlexNet and ResNet took 0:0:1s and VGG took 0:0:6s for this operation.