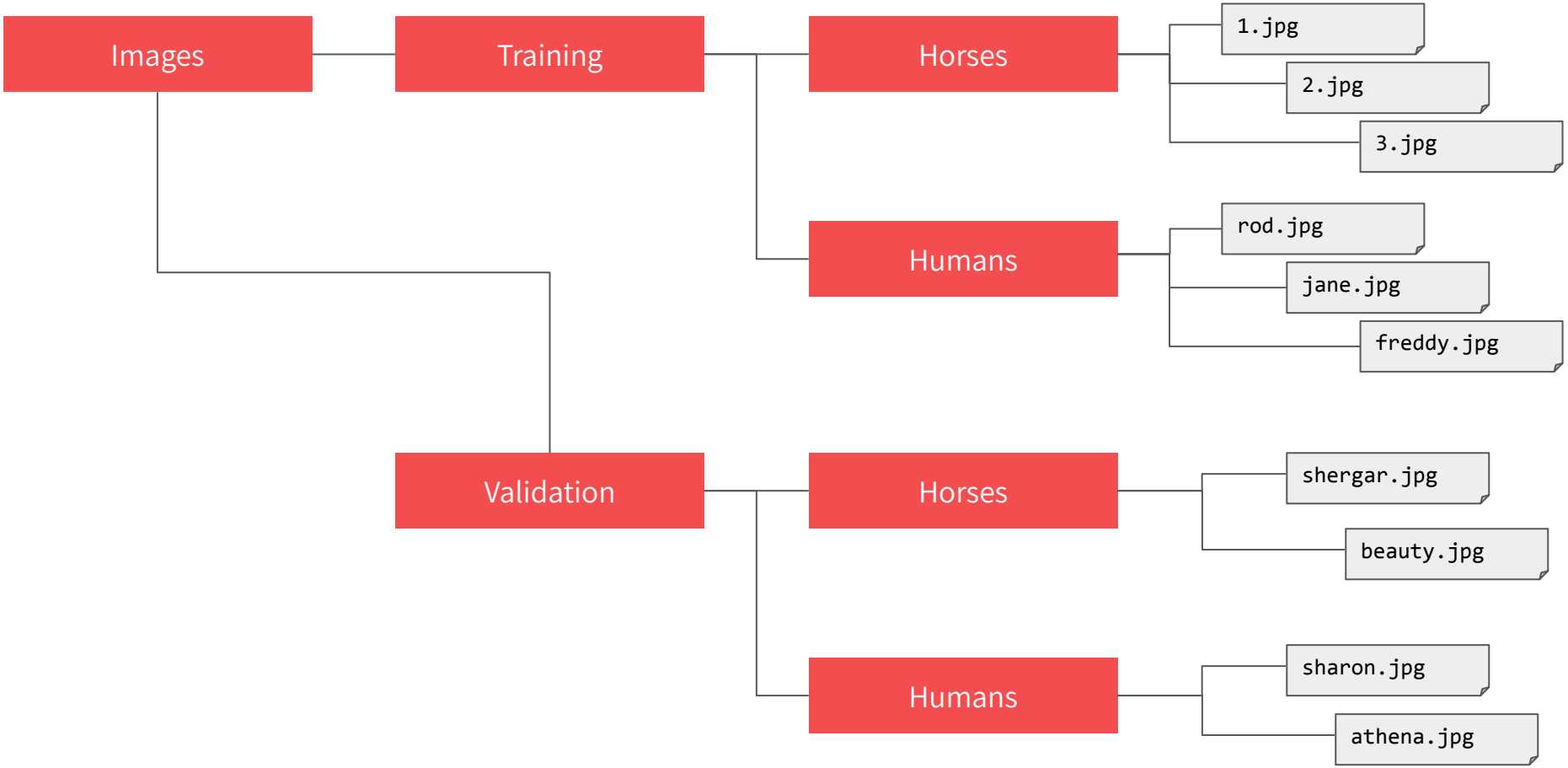
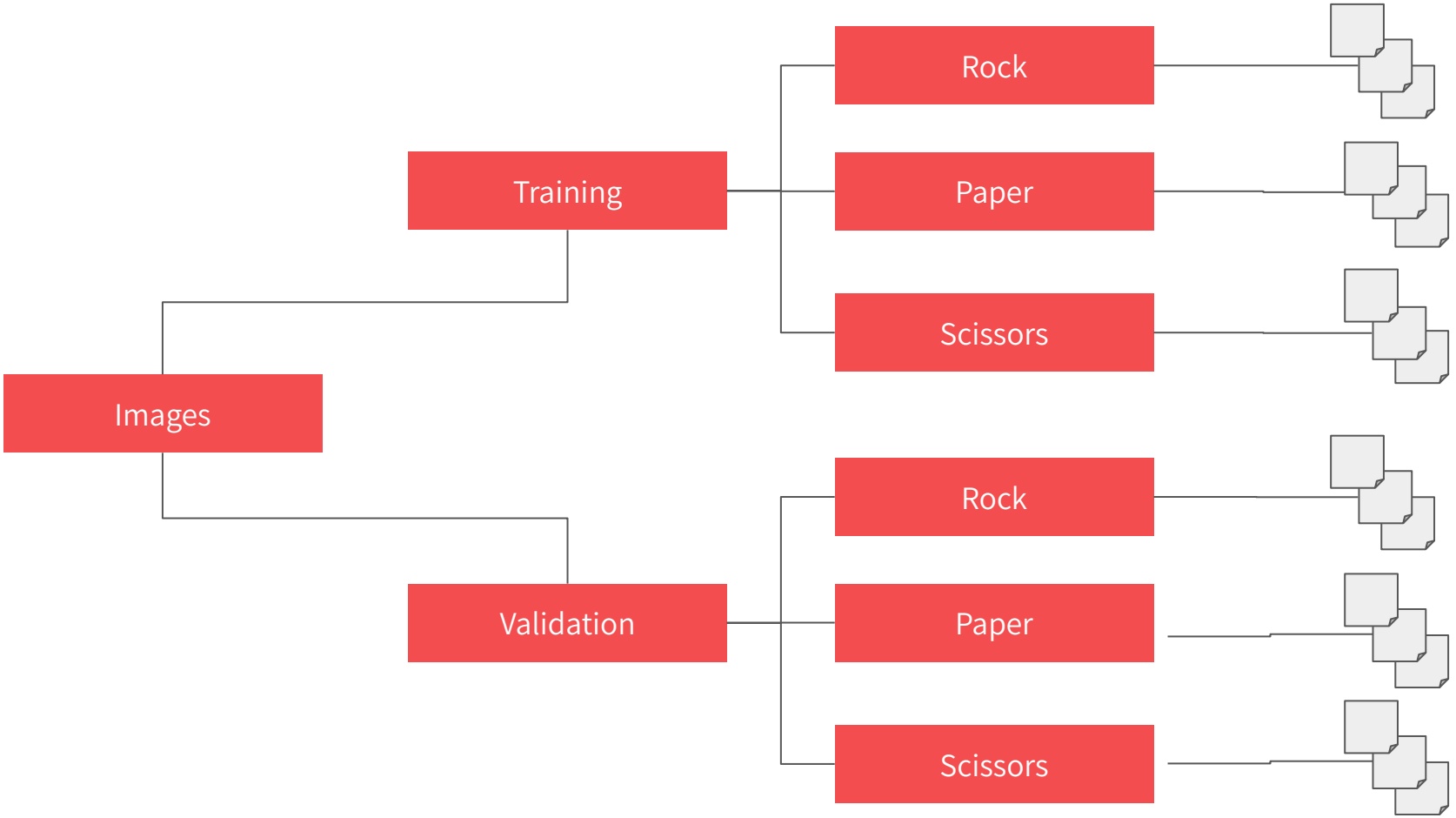
Copyright Notice


Images
Training
Horses
1.jpg
2.jpg
3.jpg
Humans
rod.jpg
jane.jpg
freddy.jpg
Validation
Horses
shergar.jpg
beauty.jpg
Humans
sharon.jpg
athena.jpg


Images
Training
Validation
Rock
Paper
Scissors
Rock
Paper
Scissors



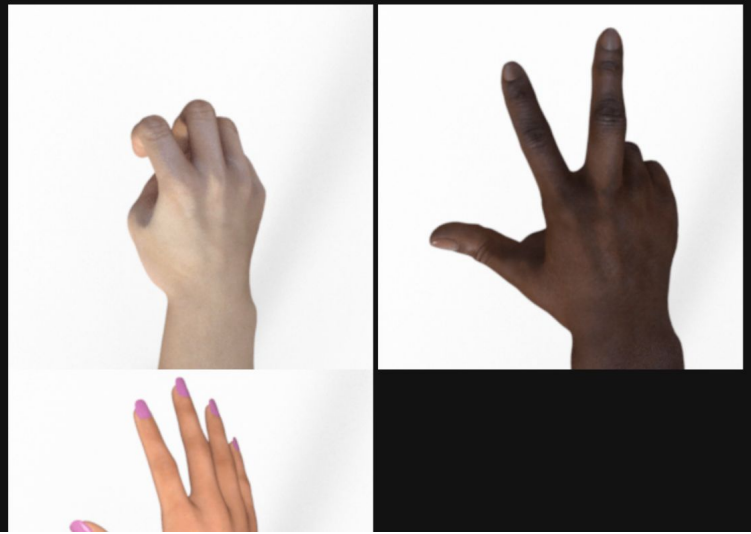
https://laurencemoroney.com/datasets.html#rock-paper-scissors-dataset

Rock Paper Scissors Dataset

Horses or Humans Dataset
Rock Paper Scissors Dataset
Yoga Poses Dataset

Rock Paper Scissors contains images from various hands, from different races, ages, and genders, posed into Rock / Paper or Scissors and labeled as such. You can download the **training set here** and the **test set here**. I created these images using CGI techniques as an experiment in determining if a model trained on a CGI-based dataset could classify real images. I also generated a few pictures that you can use for predictions. You can **find them here**.

Note that all of these pictures use a plain white background. Each image is 300×300 pixels in 24-bit color. Examples Here are a few examples showing some of the poses and the diversity of hands used.



```
train_dataset = tf.keras.utils.image_dataset_from_directory(
    train_dir,
    image_size=(150, 150),
    batch_size=20,
    label_mode='binary')
```

```
train_dataset = tf.keras.utils.image_dataset_from_directory(
    train_dir,
    image_size=(150, 150),
    batch_size=20,
    label_mode='categorical')
```

```
model = tf.keras.models.Sequential([
    tf.keras.Input(input_shape=(150, 150, 3)),
    tf.keras.layers.Conv2D(16, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(1, activation='sigmoid')
])
```

```
model = tf.keras.models.Sequential([
    tf.keras.Input(input_shape=(150, 150, 3)),
    tf.keras.layers.Conv2D(16, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(32, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.Dense(3, activation='softmax')
])
```

Rock: 0.001 Paper: 0.647 Scissors: 0.352

```
model.compile(loss='binary_crossentropy',
    optimizer=tf.keras.optimizers.RMSprop(learning_rate=0.001),
    metrics=['accuracy'])
```

```
model.compile(loss='categorical_crossentropy',
    optimizer=tf.keras.optimizers.RMSprop(learning_rate=0.001),
    metrics=['accuracy'])
```


Training and validation accuracy
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
100
Training accuracy
Validation accuracy