```
# -*- coding: utf-8 -*-
"""Evaluation accuracy and losses
Automatically generated by Colaboratory.
Original file is located at
    https://colab.research.google.com/drive/1W2GXK6PuYi3lFP0a2OUtOUB4U0xVs_BG
"""

print(h.history.keys())
histories_acc.append(h.history['acc'])
histories_val_acc.append(h.history['val_acc'])
histories_loss.append(h.history['loss'])
histories_val_loss.append(h.history['val_loss'])

histories_acc=np.array(histories_acc)
histories_val_acc=np.array(histories_val_acc)
histories_loss=np.array(histories_loss)
histories_val_loss=np.array(histories_val_loss)

print('histories_acc',histories_acc,'histories_loss',histories_loss,'histories_val_acc',histories_val_acc,'
histories_val_loss',histories_val_loss)

predictions=model.predict_proba([X_test[image_number].reshape(1,224,224,3)])

for idx,result,x in zip(range(0,6),found,predictions[0]):
  print("Label:{},Type:{},Species:{},Score:{}%".format(idx,result[0],result[1],round(x*100,3)))

ClassIndex=model.predict_classes([X_test[image_number].reshape(1,224,224,3)])

ClassIndex
print(found[ClassIndex[0]])
```

```
image_number=np.random.randint(0,len(X_test))
print(image_number)


plt.figure(figsize=(8,8))
plt.imshow(X_test[image_number])
```