GALAXY INTERNATIONAL INTERDISCIPLINARY RESEARCH JOURNAL (GIIRJ)
ISSN (E): 2347-6915
Vol. 11, Issue 11, November (2023)

OBJECT RECOGNITION IN IMAGES USING ARTIFICIAL INTELLIGENCE AND PYTHON

Солиев Бахромжон Набижонович,
Ассистент Ферганского филиала Ташкентского университета информационных технологий имени Мухаммада аль-Хорезми,
bahromjonsoliev@gmail.com

Эрматова Зарина Кахрамоновна,
Ассистент Ферганского филиала Ташкентского университета информационных технологий имени Мухаммада аль-Хорезми,

ANNOTATION

This article explores methods and technologies for recognizing objects in images using artificial intelligence (AI) and the Python programming language. We will consider various approaches to solving the problem of object recognition, present a literature review and offer practical examples of software solutions.

Keywords: Artificial intelligence, object recognition, Python, computer vision, machine learning.

INTRODUCTION

Recognizing objects in images is an important task in the field of computer vision and artificial intelligence. This has a wide range of applications, including automatic face detection, vehicle detection on the roads, medical image classification, and more. In this article, we'll look at how to use the Python programming language to solve this problem.

Literature review and methodology. To solve the problem of object recognition, we use machine learning methods, in particular, deep neural networks. This article discusses the following steps:

Data collection: We need labeled datasets that contain images with the objects we want to recognize.

Data pre-processing: We pre-process the images, including resizing, normalizing, and reducing noise.

Model selection: We choose a deep neural network, such as Convolutional Neural Network (CNN), to train our model.

Train the model: We train the model on our data using various optimizers and loss functions. Artificial intelligence (AI) training involves training a model on a large amount of data in order to create the model's ability to generalize and make predictions on new, previously unseen data. Here's the general process of AI training:

Data collection: You need to create a dataset that reflects the variety of scenarios and variations that the model may encounter in the real world.

Data preparation: Data typically undergoes pre-processing, such as scaling, normalization, and augmentation, to enable more efficient training.

Model Architecture Selection: The model architecture that is appropriate for the task at hand is selected. For example, a convolutional neural network (CNN) can be used for an image recognition task.

Definition of the loss function and optimizer: The loss function determines how well the model performs the task, and the optimizer is used to minimize this function.

Train the model: The model is trained on training data, where the inputs and their corresponding outputs are used to adjust the parameters of the model.

Hyperparameter validation and tuning: The model is validated on data it has not previously seen (validation) and the model's hyperparameters are tuned for better performance.

Performance Assessment: The model is evaluated on test data to verify its ability to generalize to new data.

Tuning and optimization: Depending on the test results, changes can be made to the model or its hyperparameters to achieve better performance.

Deployment: Once successfully trained and evaluated, the model can be deployed for real-world use.
AI training is computationally intensive, data-intensive, and often time-consuming. The training process can be repeated several times for optimal results.

Results: After training the model, we evaluate its performance on a test dataset. Examples of code for training the model and using it for object recognition will be presented in this article. Below are Python code examples for some of the object recognition steps using the TensorFlow library:

1. Installing Libraries:

```
# Installing TensorFlow and TensorFlow Hub
!pip install tensorflow tensorflow-hub
```

2. Importing Libraries:

```
import tensorflow as tf
import tensorflow_hub as hub
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input, decode_predictions
```

```
import numpy as np
```

3. Loading the pre-trained model:

```
# Download the pre-trained MobileNetV2 model
model = tf.keras.applications.MobileNetV2(weights='imagenet')
```

4. Image Pre-Processing:

```
# Image Loading & Preprocessing
img_path = 'path_to_your_image.jpg'
img = image.load_img(img_path, target_size=(224, 224))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
img_array = preprocess_input(img_array)
```

5. Predicting the object in the image:

```
# Getting Predictions
predictions = model.predict(img_array)
decoded_predictions = decode_predictions(predictions)
print("Predictions:", decoded_predictions[0])
```

These examples demonstrate the basic steps that can be used to recognize objects in images using TensorFlow and pretrained models. Please note that for full recognition of objects on real data, additional adjustment and optimization of the model will be required, depending on the specific task.

CONCLUSION

Artificial intelligence and Python provide powerful tools to solve the problem of recognizing objects in images. We've covered the key steps in this process, including data collection, preprocessing, model selection, and training. Our case studies demonstrate how to put these techniques into practice. This article can serve as a starting point for exploring more complex and interesting applications in the field of object recognition using AI and Python.

LITERATURE

1. Asrayev M. 0-SEQUENT FUNCTIONS METHODS FOR SIMPLE MEZONS IN THE FORM OF A SEX FUNCTIONALITY – 2023. – T. 1. – No. 2. – C. 9-1
2. Samijonov A. et al. Gradient method for determining non-informative features on the basis of a homogeneous criterion with a positive degree //IOP Conference Series: Materials Science and Engineering. – IOP Publishing, 2020. – T. 919. – №. 4. – C. 042011.
3. Python-Powered E-Commerce Solutions in Uzbekistan. (2023). Journal of Technical Research and Development, 1(1), 51-54. https://jtrd.mcdir.me/index.php/jtrd/article/view/3

4. Python's Role in Revolutionizing E-Commerce in Uzbekistan. (2023). Journal of Technical Research and Development, 1(1), 51-54. https://jtrd.mcdir.me/index.php/jtrd/article/view/4
5. Uzbekistan's Digital Market: Python's E-Commerce Impact. (2023). Journal of Technical Research and Development, 1(1), 58-61. https://jtrd.mcdir.me/index.php/jtrd/article/view/5
6. Navigating the E-Commerce Landscape in Uzbekistan with Python. (2023). Journal of Technical Research and Development, 1(1), 46-50. https://jtrd.mcdir.me/index.php/jtrd/article/view/1
7. Elevating E-Commerce in Uzbekistan with Python. (2023). Journal of Technical Research and Development, 1(1), 43-45. https://jtrd.mcdir.me/index.php/jtrd/article/view/2
8. Zulunov R. M., Ermatova Z. Q. PYTHON AND TENSORFLOW TECHNOLOGY FOR CREATING A ARTIFICIAL INTELLECTUAL PROGRAM WITH THE HELP OF THE JOURNAL OF Integrated Education and Research. – 2023. – T. 2. – No. 3. – C. 53-56.
9. Kayumov A. et al. PYTHON DASTURLASH TILIDA RASMLAR BILAN ISHLASH. PILLOW MODULI //Research and implementation. – 2023 4. Ermatova Z. ALTERNATIVE APPROACHES TO ERROR HANDLING: A COMPARISON OF EXCEPTIONS AND STD:: EXPECTED IN C++ // Descendants of Al-Farghani. – 2023. – T. 1. – №. 2. P. 67-73. 10. Zulunov R. M., Ermatova Z. Q. PYTHON VA TENSORFLOW YORDAMIDA SUN'IY INTELLEKT DASTURINI YARATISH TEXNOLOGIYASI //Journal of Integrated Education and Research. – 2023. – T. 2. – №. 3. P. 53-56.
11. Urinboev Abdushukur Abdurakhimovich. (2023). The Vital Role of Web Programming in the Digital Age. Journal of Science-Innovative Research in Uzbekistan, 1(6), 42–51. Retrieved from https://universalpublishings.com/index.php/jsiru/article/view/1933
12. O'rinboev A. ANALYZING THE EFFICIENCY AND PERFORMANCE OPTIMIZATION TECHNIQUES OF REACT. JS IN MODERN WEB DEVELOPMENT // Innovative Research in the Modern World: Theory and Practice. – 2023. – T. 2. – №. 24. P. 54-57.
13. Abdullayev A., Kadirova L. A. INNOVATIVE APPROACH TO FORECASTING AND MANAGING THE EMPLOYMENT OF GRADUATES OF HIGHER EDUCATION INSTITUTIONS //Экономика: анализы и прогнозы. – 2021. – №. 2. – С. 6-12.
14. Abdullaev A., Kadirova L. A., Turgunov M. A. Tekhnologiya primeneniya elektronnykh sredstva obucheniya v sisteme regional'nogo obrazovaniya [Technology of application of electronic means of teaching in the system of regional education]. – 2018. P. 120-123.
15. Sodikova M. EFFECTIVE METHODS OF TEACHING HISTORY // SCIENCE AND TECHNOLOGY. WORLD STUDIES. – 2020. P. 29-31.
16. Sadikova M. OPTIMIZATION OF THE BUSINESS PROCESS AS ONE OF THE MAIN TASKS IN MODERN MANAGEMENT. – 2022. – №. 9 (87). P. 3-7.
17. Sadikova Munira Alisherovna. (2023). DISADVANTAGES OF TEACHING PROGRAMMING IN DISTANCE EDUCATION. Intent Research Scientific Journal, 2(10), 26–33.