

УДК 004.93'1

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ БИБЛИОТЕК РАСПОЗНАВАНИЯ ЛИЦС.Н. ПЕТРОВ¹, А.Д. МАТЮШЕНКО², Д.А. РУДЕНЯ²*1 – Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь**2 – Национальный детский технопарк, Республика Беларусь**Поступила в редакцию 20 марта февраля 2023*

Аннотация. Приведены результаты сравнения эффективности библиотек распознавания лиц Face-Recognition, Dlib, DeepFace на изображениях базы Labeled Faces in the Wild (LFW). Показано, что библиотека Face-Recognition является лучшим выбором в сравнении с Dlib и DeepFace.

Ключевые слова: распознавание лиц, сверточные нейронные сети, биометрия, Face-Recognition, Dlib, DeepFace.

Введение

Распознавание лиц, это технология, которая позволяет автоматически идентифицировать или верифицировать личность человека по его изображению. Распознавание человека по изображению лица имеет ряд преимуществ по сравнению с другими методами, а именно [1], не требует специального дорогостоящего оборудования для реализации и является бесконтактным. Для большинства приложений достаточно видеокамеры для получения изображения и компьютера для его обработки. Результаты многих исследований показали [2, 3], что сверточные нейронные сети являются эффективным средством для распознавания лиц. Соответственно, появилось множество фреймворков и библиотек для создания и тренировки нейронных сетей под различные задачи. В работе проводится сравнительный анализ эффективности популярных библиотек распознавания лиц Face-Recognition, Dlib, DeepFace.

Библиотеки распознавания лиц

Библиотека Face-Recognition, основанная на библиотеке Dlib, позволяет распознавать лица на изображениях и в потоке видео. Основная особенность этой библиотеки – это ее скорость и точность. Face-Recognition использует множество методов для обработки изображений и выделения лиц, включая поиск лиц по признакам, детектирование ключевых точек на лицах, а также использование глубоких нейронных сетей для классификации лиц. Благодаря этим методам, Face-Recognition обеспечивает высокую точность распознавания, а также высокую производительность.

Библиотека Dlib также широко используется для распознавания лиц и имеет ряд преимуществ. Она использует современные методы машинного обучения, включая SVM-классификаторы и глубокие нейронные сети. Кроме того, она имеет свой собственный формат файла для хранения обученных моделей, что делает ее более удобной для использования.

Библиотека DeepFace является еще более новой и использует глубокие нейронные сети для распознавания лиц. Она предоставляет набор предобученных моделей для классификации лиц, которые можно использовать для быстрого и точного распознавания. Однако, из-за сложности и объемности глубоких нейронных сетей, производительность DeepFace может быть значительно ниже, чем у более легковесных библиотек.

Рассмотрим подробнее методы, используемые в библиотеках для распознавания лиц.

В библиотеке Dlib для распознавания лиц используется модель Dlib_face_recognition_resnet_model_v1. Эта модель использует глубокие нейронные сети для извлечения высокоуровневых признаков лиц, таких как расположение глаз, носа и рта. Затем, на основе этих признаков, модель генерирует уникальный вектор, называемый «вектором признаков», который представляет собой компактное числовое представление лица. Для сравнения лиц в Dlib используется евклидово расстояние между векторами признаков. Чем меньше расстояние между двумя векторами, тем больше вероятность, что они соответствуют одному и тому же лицу.

В библиотеке Face-Recognition для сравнения лиц используется метод compare_faces. Этот метод также использует вектора признаков, сгенерированные с помощью глубоких нейронных сетей. Он принимает два вектора в качестве входных данных и возвращает значение True или False, в зависимости от того, соответствуют ли они одному и тому же лицу. Для сравнения лиц в методе compare_faces также используется евклидово расстояние между векторами признаков.

В библиотеке DeepFace для верификации лиц используется метод verify. Этот метод принимает два изображения и определяет, соответствуют ли они одному и тому же лицу. Для этого он использует сверточные нейронные сети, предобученные на большом наборе данных лиц. Он генерирует векторы признаков для каждого изображения и затем вычисляет расстояние между ними. Если расстояние меньше заданного порога, метод verify вернет значение True, что означает, что изображения соответствуют одному и тому же лицу.

Таким образом, евклидово расстояние между векторами признаков является ключевым элементом для сравнения и верификации лиц в библиотеках для распознавания лиц. Оно позволяет быстро и точно определять, соответствуют ли два изображения одному и тому же лицу, основываясь на вычислении расстояния между векторами. Благодаря этому подходу, эти библиотеки показывают высокую точность распознавания лиц.

Для исследования эффективности указанных библиотек была использована база данных фотографий лиц Labeled Faces in the Wild (LFW) [4]. Эта база содержит 13233 изображения 5749 человек, которые были собраны в Интернете. Каждое изображение имеет размер 250×250 пикселей в формате jpg.

Результаты исследования эффективности библиотек

Код, написанный на языке Python для реализации каждого из методов, продемонстрирован на рис. 1–3.

```
import os
import shutil
import face_recognition

dir_list = os.listdir("C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled")
T = 0
N = 0
R = 0

for folder_name in dir_list:
    file_names = os.listdir(
        f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}")
    R += 1
    if R % 100 == 0:
        print(T / N)
        print(str(R) + "-----")
    if len(file_names) >= 2:
        N += 1
        pic1 = face_recognition.load_image_file(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[0]}")
        try:
            pic1_encode = face_recognition.face_encodings(pic1)[0]
        except IndexError:
            continue
        pic2 = face_recognition.load_image_file(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[1]}")
        try:
            pic2_encode = face_recognition.face_encodings(pic2)[0]
        except IndexError:
            continue
        results = face_recognition.compare_faces([pic1_encode], pic2_encode)
        print(N)
        if results[0] == True:
            T += 1
    print(T / N)
```

Рис. 1. Код для тестирования метода распознавания лиц на основе библиотеки Face-Recognition

```

from deepface import DeepFace
import cv2
import os
import shutil

dir_list = os.listdir("C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled")
T = 0
N = 0
R = 0

for folder_name in dir_list:
    file_names = os.listdir(
        f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}")
    R += 1
    if R % 100 == 0:
        print(T / N)
        print(str(R) + "-----")
    if len(file_names) >= 2:
        N += 1
        img1 = cv2.imread(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[0]}")
        img2 = cv2.imread(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[1]}")
        try:
            results = DeepFace.verify(img1, img2)
        except ValueError:
            continue
        print(N)
        if results['distance'] < 0.6:
            T += 1
    print(T / N)

```

Рис. 2. Код для тестирования метода распознавания лиц на основе библиотеки DeepFace

```

import dlib
import numpy as np
import os
import shutil
import face_recognition

dir_list = os.listdir("C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled")
T = 0
N = 0
R = 0

detector = dlib.get_frontal_face_detector()
sp = dlib.shape_predictor("shape_predictor_5_face_landmarks.dat")
facerec = dlib.face_recognition_model_v1("dlib_face_recognition_resnet_model_v1.dat")

for folder_name in dir_list:
    file_names = os.listdir(
        f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}")
    R += 1
    if R % 100 == 0:
        print(T / N)
        print(str(R) + "-----")
    if len(file_names) >= 2:
        N += 1
        img1 = dlib.load_rgb_image(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[0]}")
        img2 = dlib.load_rgb_image(
            f"C:/Users/fluffy11/PycharmProjects/pythonProject3/reco_test/lfw-deepfunneled/{folder_name}/{file_names[1]}")
        img1_detection = detector(img1, 1)
        img2_detection = detector(img2, 1)
        try:
            img1_shape = sp(img1, img1_detection[0])
            img2_shape = sp(img2, img2_detection[0])
        except IndexError:
            continue

        img1_aligned = dlib.get_face_chip(img1, img1_shape)
        img2_aligned = dlib.get_face_chip(img2, img2_shape)
        img1_representation = facerec.compute_face_descriptor(img1_aligned)
        img2_representation = facerec.compute_face_descriptor(img2_aligned)
        img1_representation = np.array(img1_representation)
        img2_representation = np.array(img2_representation)
        euclidean_distance = img1_representation - img2_representation
        euclidean_distance = np.sum(np.multiply(euclidean_distance, euclidean_distance))
        euclidean_distance = np.sqrt(euclidean_distance)

        print(N)
        if euclidean_distance < 0.6:
            T += 1
    print(T / N)

```

Рис. 3. Код для тестирования метода распознавания лиц на основе библиотеки Dlib

Результат тестирования библиотек распознавания лиц представлен в табл. 1., рассчитана метрика «точность» (Accuracy).

Табл. 1. Результаты сравнения эффективности библиотек распознавания лиц

Название библиотеки	Accuracy
Face-Recognition	96,13%
DeepFace	93,63%
Dlib	96,07%

Как видно из таблицы, библиотека Face-Recognition показывает большую точность среди рассмотренных библиотек. Реализация на Dlib также показала высокую точность, но при этом программный код для распознавания оказался более массивным.

Заключение

Согласно результатам исследования методов распознавания лиц, отметим, что Face-Recognition является лучшим выбором в сравнении с Dlib и DeepFace. Причина этому заключается в том, что Face-Recognition использует оптимизированные алгоритмы, которые позволяют ей работать быстрее и эффективнее, чем другие библиотеки. Кроме того, Face-Recognition поддерживает не только распознавание лиц, но и обнаружение, а также выравнивание лиц, что делает ее полноценным инструментом для работы с изображениями лиц.

В целом, использование библиотек для распознавания лиц является важным инструментом для автоматической идентификации людей на изображениях и в видеопотоках. Благодаря высокой точности и быстродействию, эти библиотеки могут быть использованы в различных областях, таких как безопасность, медицина, социальные сети и т.д.

STUDY OF THE EFFECTIVENESS OF FACE RECOGNITION LIBRARIES

S.N. PETROV, A.D. MATSIUSHENKA, D.A. RUDENYA

Abstract. The results of comparing the effectiveness of face recognition libraries Face-Recognition, Dlib, Deep Face on the images of the Labelled Faces in the Wild (LFW) database are presented. It is shown that the Face-Recognition library is the best choice in comparison with Dlib and DeepFace.

Keywords: face recognition, convolutional neural networks, biometrics, Face-Recognition, Dlib, DeepFace.

Список литературы

1. Антончик, А.В. // Докл. БГУИР. 2009. № 2 (40). С. 67–72.
2. Мищенко Е.С. // Вестник Волгоградского государственного университета. Серия 9. Исследования молодых ученых. 2016. № 11. С. 74–76.
3. Паршин С.Е. // Сборник научных трудов НГТУ. 2019. № 1 (94). С. 55–70.
4. Labelled Faces in the Wild (LFW) Dataset [Электронный ресурс]. URL: <https://www.kaggle.com/datasets/jessicali9530/lfw-dataset>.