В настоящее время желательно использовать комбинацию перечисленных методов по устранению угрозweb-приложений.

ЛИТЕРАТУРА

1. Урбанович, П. П. Защита информации: конспект-лекция, ч. 2 = Information Protection, Part 2: BASIC METHODS / П. П. Урбанович. – Минск: БГТУ, 2019. – 34 с. – на англ. яз.

2. Cross-site Scripting (XSS): определение и предотвращение [Электронный ресурс – Режим доступа: https:// webdevblog.ru/cross-site-scripting-xss-opredelenie-i-predotvrashhenie/. – Дата доступа: 13.04.2023.

3. Предотвращение атак CSRF (Laravel 8.x) –Laravel Framework Russian Community [Электронный ресурс].– Режим доступа: https://laravel.su/docs/8.x/csrf– Дата доступа: 18.04.2023.

УДК 004.054.53

Студ. С.Д. Рудаковский, Е.И. Сапегина
Науч. рук. проф. П.П. Урбанович
(Кафедра информационных систем и технологий, БГТУ)

СРАВНЕНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ БИБЛИОТЕК, РЕАЛИЗУЮЩИХ АЛГОРИТМ РАСПОЗНАВАНИЯ ЛИЦ

Технологии распознавания лиц призваны улучшить безопасность, повысить эффективность и повысить качество жизни [1].

Распознавание лиц – это процесс автоматического определения личности на основе физических особенностей лица человека, таких как форма лица, расположение глаз, носа, рта и т.д. [2]. В связи с этим разработано большое множество алгоритмов и методы, позволяющих реализовать распознавание лиц.

Реализация алгоритмов распознавания лиц может быть достаточно сложной задачей, требующей использования специализированных библиотек и фреймворков. Существует множество библиотек и фреймворков, которые могут помочь в реализации распознавания лиц. Некоторые из них являются открытыми и бесплатными, в то время как другие могут предоставляться на коммерческой основе. Кроме того, каждая библиотека имеет свои особенности и возможности. В этом контексте рассмотрим библиотеки, реализующие алгоритмы для распознавания лиц, их основные особенности, преимущества и недостатки, сравним их производительность и точность распознавания. Для сравнения были выбраны две библиотеки: OpenCV и Dlib.

OpenCV– это библиотека компьютерного зрения с открытым исходным кодом, которая включает в себя функции для обработки

изображений и видео, включая распознавание лиц [3]. Используетал-горитмы: HaarcascadesиLocalBinaryPatternsHistograms (LBPH).

Dlib–библиотека с открытым исходным кодом, которая предоставляет высокоуровневые API для распознавания лиц, а также низкоуровневые инструменты для обработки изображений. Dlib использует модель распознавания лиц, основанную на DeepMetricLearning, которая позволяет получить высокую точность распознавания лиц [4].

Для сравнения вышеуказанных библиотек на практикемы написали 2 программы на языке Python, каждая из которых реализует процесс распознавания лиц, используя или OpenCV, или DLib [4].

Нельзя однозначно сделать вывод о эффективности работы алгоритмов на основе только написанного нами кода. Однако, можно проанализировать время выполнения алгоритмов и точность обнаружения лиц, чтобы сделать предварительные выводы. Например, если сравнивать время выполнения алгоритмов, то на больших изображениях с множеством лиц библиотека Dlib работает быстрее, так как она использует более эффективные алгоритмы и оптимизирована для работы с многопоточностью. Однако, на небольших изображениях или изображениях с одним лицом, скорость работы примерно одинаковая.

Что касается точности обнаружения лиц, то здесь многое зависит от обученной модели и параметров алгоритма. Обе библиотеки, OpenCV и Dlib, имеют высокую точность обнаружения лиц, но также могут допускать ошибки, особенно на фотографиях с плохим освещением, засветами, размытием и т.д.

Резюмируя, OpenCV использует алгоритмы Haarcascades и LocalBinaryPatternsHistograms (LBPH) для распознавания лиц. Хотя эти алгоритмы могут быть достаточно быстрыми и могут обрабатывать большие объемы данных, они могут быть менее точными, особенно при работе с изображениями с низким качеством и в неприятных условияхосвещения. Dlib, с другой стороны, использует модель глубокого обучения на основе метрического обучения (deepmetriclearning), которая позволяет получать высокую точность распознавания лиц. Однако использование этой модели может требовать больших объемов вычислительных ресурсов, таких как графические процессоры. Поэтому перед выбором конкретной библиотеки необходимо тщательно оценить свои потребности и провести тестирование для выбора наиболее подходящего решения.

ЛИТЕРАТУРА

1. Урбанович, П. П. Защитаинформации: конспект-лекция, ч.1 = Information Protection, Part 1: Introduction To The Subject Area. – Минск: БГТУ, 2019. – 52 с. https://elib.belstu.by/handle/123456789/29335

2. Facialmapping (landmarks) withDlib + python. [Электронныйресурс]. Режим доступа:https://towardsdatascience.com/facial-mapping-landmarks-with-dlib-python. Дата доступа: 06.04.2023.

3. OpenCV [Электронный ресурс]. Режим доступа: https://blog. skillfactory.ru /glossary/opencv/. Датадоступа: 05.04.2023.

4. Системы распознавания лиц. [Электронный ресурс]. Режим доступа:https://www.tadviser.ru/index.php/Facial_recognition. Дата доступа: 10.04.2023.

УДК 004.056.5:004.021

Студ. А.В. Почебут, В.В. Максимова
Науч. рук. проф. П.П. Урбанович
(Кафедра информационных систем и технологий, БГТУ)

СРАВНИТЕЛЬНАЯ ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ И КРИПТОСТОЙКОСТИ АЛГОРИТМОВ RC4 И RC4+

Алгоритмы RC4 и RC4+ играют важную роль в повышении защищенности информации [1]. Ключевая разница между RC4 и RC4+ в дополнительном блоке EKSA, который повышает криптографическую стойкость алгоритма. RC4 состоит из двух блоков: KSA (Key-Scheduling Algorithm), производящего обработку ключа для генерации псевдослучайной последовательности ключей (до 256 бит), а также PRGA (Pseudo-Random Generation Algorithm), выполняющего генерацию псевдослучайной последовательности ключей. В RC4+ EKSA (Extended Key-Scheduling Algorithm) блок идет следом за KSA блоком [1–3].

EKSA – это дополнительный блок в алгоритме KSA в RC4+, который улучшает безопасность шифра путем добавления дополнительных итераций в KSA, используя два ключа длиной 128 битов каждый. Этот блок делает дополнительные шаги обработки ключа перед генерацией ключевого потока, которые включают в себя дополнительные перестановку в KSA блоке и добавление к ключу дополнительного блока битов. Это повышает стойкость алгоритма и уменьшает вероятность успешной атаки на систему шифрования. При использовании коротких ключей (например, менее 128 бит) применение EKSA может привести к увеличению количества обращений к комбинации, содержащей оригинальный ключ, необходимой для генерации псевдослучайной последовательности ключей, что в свою очередь может повысить вероятность обнаружения ключевого потока и ухудшить стойкость алгоритма. Однако, при использовании достаточно длинных ключей, применение EKSA может повысить криптостойкость алгоритма RC4+.