

DOI: 10.12731/2070-7568-2022-11-1-53-67

УДК 004.85

РАЗРАБОТКА МОДУЛЯ АВТОМАТИЧЕСКОГО ОБНАРУЖЕНИЯ ОБЛАСТИ ГЛАЗ ДЛЯ ПРИМЕНЕНИЯ В АВТОМАТИЗИРОВАННОЙ СИСТЕМЕ ТЕЛЕМЕДИЦИНСКИХ КОНСУЛЬТАЦИЙ

С.О. Бурдуковский

Работа посвящена разработке модуля автоматического обнаружения области глаз и является частью научного исследования, посвященного разработке информационной системы автоматического диагностирования косоглазия, которая будет применена в автоматизированной системе телемедицинских консультаций. Телемедицина, имеющая большой потенциал для снижения растущего спроса на диагностику офтальмологических заболеваний, является эффективным методом для своевременного выявления косоглазия. Для достижения цели были решены следующие задачи: обучение модели обнаружения области глаз, разработка программного интерфейса приложения для работы с моделью обнаружения, разработка веб-интерфейсов пациента и администратора. Обученная модель может применяться в различных информационных системах, в которых требуется функционал обнаружения глаз. Также результаты могут быть использованы в различных исследованиях, в которых применяется платформа TensorFlow для обнаружения объектов одного класса. Представленная архитектура взаимодействия модели обнаружения, программного интерфейса приложения и веб-интерфейсов может быть использована при проектировании и разработке сложных систем, включающих пользовательские интерфейсы и вычисления на серверной стороне.

Цель – разработка модуля автоматического обнаружения области глаз на фотографии лица человека, направленного на использование в автоматизированной информационной системе телемедицинских консультаций.

Метод или методология проведения работы: в статье использовались методы программирования и машинного обучения.

Результаты: разработан модуль автоматического обнаружения области глаз для применения в автоматизированной системе телемедицинских консультаций.

Область применения результатов: полученные результаты целесообразно применять при организации автоматизированных телемедицинских услуг и для внедрения в разрабатываемые информационные системы, использующие телемедицинские технологии.

Ключевые слова: телемедицина; обнаружение объектов; информационная система; машинное обучение

DEVELOPMENT OF AUTOMATIC EYE DETECTION MODULE FOR USE IN AUTOMATED SYSTEM OF TELEMEDICAL CONSULTATIONS

S.O. Burdukowsky

This work is devoted to development of automatic eye detection module. This is a part of a scientific study devoted to development of automatic strabismus detection information system, which will be applied in automated system of telemedical consultations. Telemedicine has great potential to reduce the growing demand for the diagnosis of ophthalmic diseases and is an effective method for timely detection of strabismus. The following tasks were solved to achieve the goal: training of eye detection model, development of an application programming interface for interacting with detection model, development of web interfaces for patient and administrator. The trained model can be used in various information systems that require eye detection functionality. Also, results can be applied to various studies that use the TensorFlow platform to detect single class objects. Presented architecture of interaction between detection model, application programming interface and web interfaces can be used in design and development of complex systems that include user interfaces and server-side computing.

Purpose. *Development of automatic module for eye detection in human face photo aimed at use in automated system of telemedical consultations.*

Methodology: *programming and machine learning methods were used in the article.*

Results: *automatic eye detection module for use in automated system of telemedical consultations was developed.*

Practical implications *it is expedient to apply the received results for organization of automated telemedicine services and implementation in the developed information systems using telemedicine technologies.*

Keywords: *telemedicine; object detection; information system; machine learning*

Введение

Телемедицина – это предоставление услуг здравоохранения в условиях, когда расстояние является критическим фактором, работниками здравоохранения, использующими информационно-коммуникационные технологии для обмена необходимой информацией в целях диагностики, лечения и профилактики заболеваний и травм, проведения исследований и оценок, а также для непрерывного образования медицинских работников в интересах улучшения здоровья населения и развития местных сообществ [2].

Разработанный модуль автоматического обнаружения области глаз является частью информационной системы диагностики косоглазия, которая будет применена в автоматизированной системе телемедицинских консультаций. Косоглазие является одним из наиболее важных офтальмологических заболеваний в жизни человека. Своевременное выявление косоглазия способствует его прогнозированию и лечению. Телемедицина, имеющая большой потенциал для снижения растущего спроса на диагностику офтальмологических заболеваний, является эффективным методом для своевременного выявления косоглазия.

Разработка модуля предполагает решение следующих задач:

- обучение модели обнаружения области глаз на фотографии,
- разработка веб-API для работы с моделью обнаружения по сети,

- разработка веб-интерфейса пациента,
- разработка веб-интерфейса администратора.

Обучение модели обнаружения

Было предложено множество традиционных алгоритмов для обнаружения глаз. Они обычно проектируются в соответствии с геометрическими характеристиками глаза. Их можно разделить на два класса. Первый класс – это геометрическая модель. Roberto Valenti и Theo Gevers [16] использовали кривизну изофот для разработки системы голосования для локализации глаза и зрачка. Nenad Markuš и соавторы [4] предложили метод локализации зрачка глаза на основе ансамбля рандомизированных деревьев регрессии. Fabian Timm и Erhardt Barth [15] предложили использовать градиенты изображения и квадраты скалярных произведений для обнаружения зрачков. Второй класс – это сопоставление с шаблоном. Lech Swirski и соавторы предложили метод RANSAC [14] для создания эллиптического уравнения для определения центра зрачка. Gabriel M. Araujo и соавторы [5] описали детектор внутреннего продукта для локализации глаз на основе корреляционных фильтров. Традиционные детекторы глаз иногда могут дать хорошие результаты, но они легко выходят из строя при изменении внешнего освещения или окклюзии лица.

В настоящее время эффективным методом обнаружения различных объектов на изображении признаны алгоритмы глубокого обучения. Глубокое обучение позволяет вычислительным моделям, состоящим из нескольких уровней обработки, изучать представления данных с несколькими уровнями абстракции. Эти методы значительно улучшили современный уровень распознавания речи, визуального распознавания объектов, обнаружения объектов и многих других областей, таких как открытие лекарств и геномика [10, 17].

Для решения задачи были проанализированы инструменты глубокого обучения. По результатам анализа был выбран фреймворк TensorFlow 2, как хорошо поддерживаемый, открытый и поддерживающий большое количество технологий таких, как CUDA (Compute Unified Device Architecture), автоматическое дифференцирование,

предтренированные модели, рекуррентные нейронные сети, сверточные нейронные сети, машина Больцмана, глубокая сеть доверия, вычисления на компьютерном кластере [1].

Платформа TensorFlow предоставляет комплекс предтренированных моделей обнаружения, которые можно дообучить, для детектирования каких-либо объектов, либо обучить с нуля.

В результате ряда экспериментов дообучения и обучения с нуля с разными моделями обнаружения, самую высокую оценку точности получила модель «SSD MobileNet V2 FPNLite 320×320», обученная с нуля.

Для обучения модели были взяты изображения лиц из базы данных Flickr-Faces-HQ Dataset. Это набор высококачественных изображений человеческих лиц, изначально созданный в качестве эталона для генеративных состязательных сетей [7]. Набор данных состоит из высококачественных изображений PNG с разрешением 1024 × 1024 и содержит значительные различия с точки зрения возраста, этнической принадлежности людей и фона изображения. Он также имеет хороший охват аксессуаров, таких как очки, солнечные очки, шляпы и т. д. Автором использовались миниатюры оригинальных изображений в размере 128×128 пикселей. Всего было использовано из набора данных 25000 изображений. Из всех изображений 2500 составили оценочную выборку, а 22500 – обучающую.

Количество шагов обучения – 50000. Каждый шаг обучения представляет собой обучение на 8 изображениях (размер партии данных). Это значит, что обучение было произведено на 400000 экземплярах изображений. Так как всего изображений в обучающей выборке 22500, то количество эпох обучения составило приблизительно 17,8. Размер партии данных подобран эмпирическим путем исходя из технических данных компьютера, на котором проводились эксперименты. Размер партии влияет только на производительность обучения и не влияет на результаты. Количество шагов обучения (50000) было определено также эмпирически, на основе нескольких экспериментов. В каждом эксперименте примерно на сорокатысячном шаге потери классификации и локализации стабилизируются.

вались, и значительных изменений в сторону уменьшения потерь не наблюдалось.

Для демонстрации работы обученной модели был написан скрипт на языке Python с использованием библиотеки TensorFlow. Результатом работы скрипта являются изображения с выделенными на нем областями с глазами. Пример на рис. 1. Проценты обозначают степень соответствия найденных объектов набору признаков обнаружения модели.

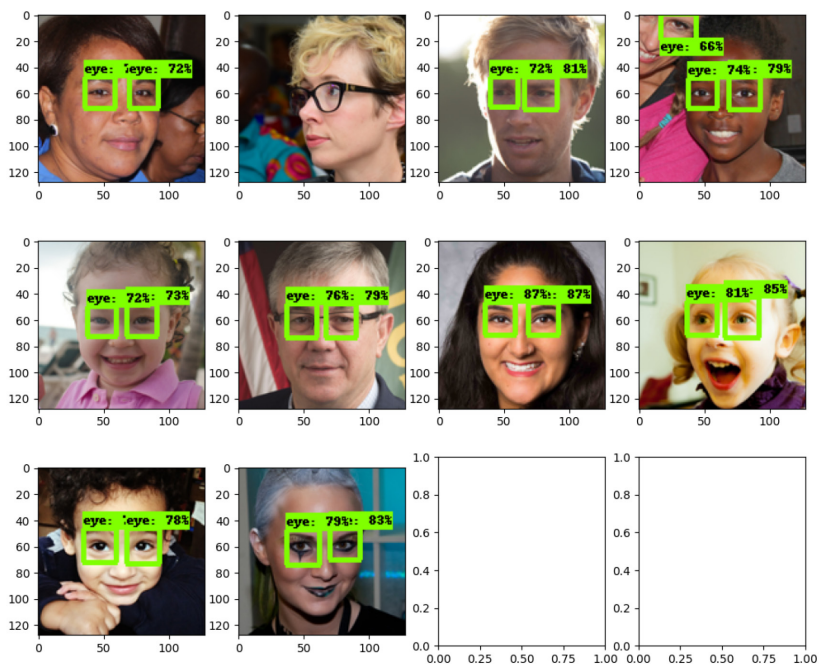


Рис. 1. Демонстрация работы обученной модели

В качестве метрики для модели обнаружения объектов используется значение IoU, что значит Intersection over Union. Оно обозначает отношение площади пересечения оригинальной рамки объекта на изображении и рамки объекта, найденной моделью, к площади области, полученной путем объединения этих рамок. Для оценки точности работы детектора объектов берется среднее IoU из отрез-

ка $[0,50; 0,95]$ с шагом 0,05. То есть среднее арифметическое долей обнаруженных объектов с IoU больше 50%, 55%, 60% и так далее до 95%. Обученная модель получила оценку 0,695. Для конкретных значений IoU точность обнаружения составила: для IoU 0,50 – 99,9%, для 0,55 – 99,5%, для 0,60 – 98,8%, для 0,65 – 97,4%, для 0,70 – 94,5%, для 0,75 – 89,8%. Согласно исследованиям [8, 9, 11-13] оценка IoU больше 0,5 обычно считается хорошим результатом.

Разработка веб-API

Для разработки веб-API был выбран веб-фреймворк Flask. Выбор был сделан в пользу этого веб-фреймворка, потому что он имеет открытый исходный код, активно поддерживается и имеет большое сообщество разработчиков. Также этот микрофреймворк имеет очень маленький размер и сознательно предоставляет лишь самые базовые возможности, что обеспечивает высокую производительность работы [6]. В качестве языка программирования выбран Python, так как модель обнаружения области глаз также разрабатывается на этом языке программирования с использованием платформы TensorFlow, и выбор Python позволит объединить веб-API и модель обнаружения в одну информационную систему.

API помимо того, что служит для обработки HTTP-запросов от клиентских приложений с одной стороны и для вызова модели обнаружения области глаз с другой, еще производит журналирование HTTP-запросов и сохранение результатов работы модели обнаружения в базе данных PostgreSQL.

Разработанный API содержит два контроллера обработки HTTP-запросов. Первый – MainController – обрабатывает запросы от клиентского приложения пациента, в котором пользователь загружает фотографию глаз для дальнейшего анализа. Второй – AdminController – обрабатывает запросы, которые осуществляет клиентское приложение администратора информационной системы. Эти запросы служат для предоставления информации о версии API, версии модели обнаружения, просмотра журналов использования API и просмотра информации о результатах обнаружения, сохраненных в базе данных PostgreSQL.

MainController предоставляет POST-запрос «recognize», который принимает фотографию глаз человека. При обработке запроса принятая фотография отправляется модели обнаружения. В зависимости от статуса работы модели обнаружения формируется ответ в формате JSON, который содержит статус обработки запроса («успешно» или «не успешно») и информацию о координатах областей глаз. Если сервис распознавания не смог обнаружить глаза на фотографии, то возвращается соответствующее сообщение и пользователю нужно будет повторить отправку. После обработки запроса выполняется сохранение результата работы модели обнаружения и текущее время в базу данных.

AdminController предоставляет три GET-запроса: «version», «log» и «results». Запрос «version» возвращает клиенту ответ в формате JSON, который содержит объект с двумя полями: «api» и «recognitionSystem». В полях указываются версии API и модели обнаружения соответственно. Запрос «log» возвращает клиенту журнал HTTP-запросов к API. Запрос имеет два параметра – «from» и «limit», которые позволяют запросить журнал от любой даты и ограничить количество записей. Ответ на запрос «log» представляет собой JSON-массив с сущностями, описывающими запросы (поля «address», «timestamp» и «ip», содержащие адрес, время запроса и IP-адрес клиента соответственно). Запрос «results» возвращает клиенту результаты работы модели обнаружения. Этот запрос, подобно запросу «log», имеет два параметра – «from» и «limit», которые позволяют запросить результаты от любой даты и ограничить количество записей. Ответ на запрос «results» представляет собой JSON-массив с сущностями, описывающими результат работы модели распознавания (поля «response» и «timestamp» и «status», содержащие тело ответа, время проведения обнаружения и информацию об успехе обработки фотографии соответственно).

Разработка веб-интерфейсов

Для разработки клиентских частей был выбран фреймворк Angular версии 13, как открытый, хорошо поддерживаемый, име-

ющий большое сообщество разработчиков инструмент разработки пользовательских интерфейсов. Angular позволяет разрабатывать сложные информационные системы корпоративного уровня, предоставляет модульную архитектуру, поддерживает все современные браузеры. Фреймворк разработан компанией Google [3], постоянно обновляется и поддерживается, что подчеркивает его актуальность. В качестве языка программирования используется TypeScript, разработанный компанией Microsoft. TypeScript является обратно совместимым с JavaScript и компилируется в последний. TypeScript имеет строгую типизацию, что является решающим фактором при выборе этого инструмента.

Архитектура клиентских частей имеет общие черты, а именно представляет собой три модуля: AppModule, CoreModule и SharedModule. AppModule включает в себя компоненты, отвечающие за работу страниц веб-интерфейса. CoreModule включает в себя сервисы для работы с серверной частью и хранилищем данных. А SharedModule включает в себя компоненты и директивы общего назначения, которые могут использоваться в различных субмодулях AppModule.

Веб-интерфейс пациента представляет собой главный экран, экран диагностики и экран результатов диагностирования. Каждый экран со стороны архитектуры приложения представляет собой отдельный ViewModule, включенный в AppModule приложения.

На главном экране отображается приветственное сообщение и описание информационной системы. Также на этом экране имеется переход к странице диагностики.

На странице диагностики веб-приложение запрашивает браузерное разрешение на использование веб-камеры, в случае если пациент хочет сделать фотографию глаз прямо в веб-интерфейсе. Также пациент может загрузить с устройства уже готовую фотографию. На фотографии должны быть четко видны глаза. После выбора фотографии с устройства или создания снимка в браузере, картинка отправляется серверной части путем HTTP-запроса. Сервер обрабатывает изображение и в ответ на запрос выдает информацию в

формате JSON, которая содержит статус обработки запроса («успешно» или «не успешно»), информацию о координатах обнаруженных объектов. Если сервис распознавания не смог обнаружить глаза на фотографии, то возвращается соответствующее сообщение и пользователю нужно будет повторить отправку.

После успешной обработки веб-интерфейс переходит на экран результатов. На этом экране пользователь может увидеть выделенные области с глазами на изображении.

Веб-интерфейс администратора представляет собой экран входа, экран просмотра журнала работы API и экран просмотра результатов работы модели обнаружения.

Экран входа представляет собой форму входа с полями «E-mail» и «Пароль». Если пользователь введет корректные данные входа и отправит форму, то будет произведен переход к основному экрану. В случае неверных данных входа, пользователь увидит соответствующее сообщение.

Основной экран представляет собой боковое меню с двумя элементами: «Журнал» и «Результаты». По умолчанию открывается страница «Журнал». На ней администратор может просматривать журнал работы API. Изначально запрашивается 100 записей от текущей даты. При прокрутке содержимого страницы до конца, автоматически происходит запрос следующих 100 записей от даты последней загруженной записи.

На странице «Результаты» администратор может просматривать результаты работы модели обнаружения, которые содержат тело ответа, время проведения обнаружения и информацию об успехе обработки фотографии соответственно. Изначально запрашивается 20 записей, а запрос последующих записей осуществляется таким же способом, как и на странице «Журнал».

Заключение

Обученная модель может использоваться не только в информационной системе автоматического диагностирования косоглазия, разрабатываемой автором статьи, но и в других информационных

системах, в которых требуется функционал обнаружения на фотографии областей с глазами. Также результаты могут быть использованы в различных исследованиях, в которых применяется платформа TensorFlow для обнаружения объектов на изображениях, и производится обнаружение объектов только одного класса.

Серверная часть является веб-API сервисом, обслуживающим запросы клиентских частей, а клиентские части являются веб-приложениями для администраторов и пациентов. Благодаря клиент-серверной архитектуре разработанная система готова к расширению – разработке мобильных клиентов для различных операционных систем. Для разработки использовались современные, открытые, активно поддерживаемые технологии и подходы, что позволяет использовать в приложениях все актуальные возможности платформ и облегчает поддержку и дальнейшую разработку системы.

Список литературы

1. Бурдуковский С.О. Анализ и оценка программ глубинного обучения // Развитие интеллектуального потенциала молодежи Кубани – 2019: Материалы II Всероссийской научно-практической конференции. Анапа, 2019. С. 18-24.
2. Всемирная организация здравоохранения. Телемедицина. Возможности и развитие в государствах-членах. Женева: Всемирная организация здравоохранения, 2012. Т. 2. 93 с.
3. Angular (фреймворк) — Википедия, 2022. URL: [https://ru.wikipedia.org/wiki/Angular_\(фреймворк\)](https://ru.wikipedia.org/wiki/Angular_(фреймворк)) (дата обращения 24.01.2022).
4. Eye pupil localization with an ensemble of randomized trees / Markuš N., Frljak M., Pandžić I.S., Ahlberg J., Forchheimer R. // Pattern Recognition. 2014. vol. 47. pp. 578-587. <https://doi.org/10.1016/j.patcog.2013.08.008>
5. Fast eye localization without a face model using inner product detectors / Araujo G.M., Ribeiro F.M.L., Silva E.A.B., Goldenstein S.K. // 2014 IEEE International Conference on Image Processing, Paris, 2014. pp. 1366-1370. <https://doi.org/10.1109/ICIP.2014.7025273>
6. Flask (веб-фреймворк) – Википедия, 2022. URL: [https://ru.wikipedia.org/wiki/Flask_\(веб-фреймворк\)](https://ru.wikipedia.org/wiki/Flask_(веб-фреймворк)) (дата обращения 24.01.2022).

7. Flickr-Faces-HQ Dataset (FFHQ), 2022. URL: <https://github.com/NVLabs/ffhq-dataset> (дата обращения 24.01.2022).
8. How Well Do Deep Learning-Based Methods for Land Cover Classification and Object Detection Perform on High Resolution Remote Sensing Imagery? / Zhang X., Han Liangxiu, Han Lianghao, Zhu L. // Remote Sensing. 2020. vol. 12. pp. 1-29. <https://doi.org/10.3390/rs12030417>
9. Intersection over Union (IoU) for object detection – PyImageSearch, 2016. URL: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection> (дата обращения 24.01.2022).
10. LeCun Y., Bengio Y., Hinton G. Deep learning // Nature. 2015. vol. 521. pp. 436-444. <https://doi.org/10.1038/nature14539>
11. Maxwell A., Timothy A.W., Luis A.G. Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies – Part 1: Literature Review // Remote Sensing. 2021. vol. 13. pp. 1-27. <https://doi.org/10.3390/rs13132450>
12. Measuring Labelling Quality with IOU and F1 Score | by Isaac Tan | Supahands Tech Blog | Medium, 2020. URL: <https://medium.com/supahands-techblog/measuring-labelling-quality-with-iou-and-f1-score-1717e29e492f> (дата обращения 24.01.2022).
13. Object Detection -IOU-Intersection Over Union | by Ananth Sandeep | Medium, 2019. URL: <https://medium.com/@nagsan16/object-detection-iou-intersection-over-union-73070cb11f6e> (дата обращения 24.01.2022).
14. Swirski L., Bulling A., Dodgson N.A. Robust real-time pupil tracking in highly of-axis images // Proceedings of the 7th Eye Tracking Research and Applications Symposium. N.Y., 2012. pp. 173-176. <https://doi.org/10.1145/2168556.2168585>
15. Timm F., Barth E. Accurate eye centre localisation by means of gradients // VISAPP 2011 – Proceedings of the Sixth International Conference on Computer Vision Theory and Applications. Виламора, 2011. pp. 125-130.
16. Valenti R., Gevers T. Accurate eye center location through invariant isocentric patterns // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012. vol. 34. pp. 1785-1798. <https://doi.org/10.1109/TPAMI.2011.251>

17. Xie D., Zhang L., Bai L. Deep Learning in Visual Computing and Signal Processing // *Applied Computational Intelligence and Soft Computing*. 2017. vol. 2017. pp. 1-13. <https://doi.org/10.1155/2017/1320780>

References

1. Burdukowsky S.O. Analiz i otsenka programm glubinnogo obucheniya [Analysis and evaluation of deep learning programs]. *Razvitie intellektual'nogo potentsiala molodezhi Kubani – 2019: Materialy II Vserossiyskoy nauchno-prakticheskoy konferentsii* [Development of intellectual potential of the youth of the Kuban - 2019: Proceedings of the II All-Russian scientific and practical conference]. Anapa, 2019, pp. 18-24.
2. World Health Organization. *Telemeditsina. Vozmozhnosti i razvitie v gosudarstvakh-chlenakh* [Telemedicine: opportunities and developments in Member States]. Geneva: World Health Organization, 2012, vol. 2, 93 p.
3. *Angular (freymvork) — Vikipediya* [Angular (framework) — Wikipedia], 2022. URL: [https://ru.wikipedia.org/wiki/Angular_\(фреймворк\)](https://ru.wikipedia.org/wiki/Angular_(фреймворк)) (accessed January 24, 2022).
4. Eye pupil localization with an ensemble of randomized trees / Markuš N., Frljak M., Pandžić I.S., Ahlberg J., Forchheimer R. *Pattern Recognition*, 2014, vol. 47, pp. 578-587. <https://doi.org/10.1016/j.patcog.2013.08.008>
5. Araujo G.M., Ribeiro F.M.L., Silva E.A.B., Goldenstein S.K. Fast eye localization without a face model using inner product detectors. *2014 IEEE International Conference on Image Processing*, Paris, 2014. pp. 1366-1370. <https://doi.org/10.1109/ICIP.2014.7025273>
6. *Flask (veb-freymvork) – Vikipediya* [Flask (web-framework) – Wikipedia], 2022. URL: [https://ru.wikipedia.org/wiki/Flask_\(веб-фреймворк\)](https://ru.wikipedia.org/wiki/Flask_(веб-фреймворк)) (accessed January 24, 2022).
7. Flickr-Faces-HQ Dataset (FFHQ), 2022. URL: <https://github.com/NVLabs/ffhq-dataset> (accessed January 24, 2022).
8. Zhang X., Han Liangxiu, Han Lianghao, Zhu L. How Well Do Deep Learning-Based Methods for Land Cover Classification and Object Detection Perform on High Resolution Remote Sensing Imagery? *Remote Sensing*, 2020, vol. 12, pp. 1-29. <https://doi.org/10.3390/rs12030417>

9. Intersection over Union (IoU) for object detection – PyImageSearch, 2016. URL: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection> (accessed January 24, 2022).
10. LeCun Y., Bengio Y., Hinton G. Deep learning. *Nature*, 2015, vol. 521, pp. 436-444. <https://doi.org/10.1038/nature14539>
11. Maxwell A., Timothy A.W., Luis A.G. Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies – Part 1: Literature Review. *Remote Sensing*, 2021, vol. 13, pp. 1-27. <https://doi.org/10.3390/rs13132450>
12. Measuring Labelling Quality with IOU and F1 Score | by Isaac Tan | Supahands Tech Blog | Medium, 2020. URL: <https://medium.com/supahands-techblog/measuring-labelling-quality-with-iou-and-f1-score-1717e29e492f> (accessed January 24, 2022).
13. Object Detection -IOU-Intersection Over Union | by Ananth Sandeep | Medium, 2019. URL: <https://medium.com/@nagsan16/object-detection-iou-intersection-over-union-73070cb11f6e> (accessed January 24, 2022).
14. Swirski L., Bulling A., Dodgson N.A. Robust real-time pupil tracking in highly off-axis images. *Proceedings of the 7th Eye Tracking Research and Applications Symposium*. N.Y., 2012, pp. 173-176. <https://doi.org/10.1145/2168556.2168585>
15. Timm F., Barth E. Accurate eye centre localisation by means of gradients. *VISAPP 2011 – Proceedings of the Sixth International Conference on Computer Vision Theory and Applications*. Vilamora, 2011, pp. 125-130.
16. Valenti R., Gevers T. Accurate eye center location through invariant isocentric patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, vol. 34, pp. 1785-1798. <https://doi.org/10.1109/TPAMI.2011.251>
17. Xie D., Zhang L., Bai L. Deep Learning in Visual Computing and Signal Processing. *Applied Computational Intelligence and Soft Computing*, 2017, vol. 2017, pp. 1-13. <https://doi.org/10.1155/2017/1320780>

ДААННЫЕ ОБ АВТОРЕ

Бурдуковский Станислав Олегович, аспирант 3-го курса
*Новосибирский государственный университет экономики и
управления*

*ул. Каменская, 56, г. Новосибирск, 630039, Российская Феде-
рация*

burdukowsky_stas@mail.ru

DATA ABOUT THE AUTHOR

Stanislav O. Burdukowsky, 3rd year postgraduate student

Novosibirsk State University of Economics and Management

56, Kamenskaya Str., Novosibirsk, 630039, Russian Federation

burdukowsky_stas@mail.ru

SPIN-code: 5117-9200

ORCID: <https://orcid.org/0000-0003-4039-1141>

Поступила 15.03.2022

После рецензирования 25.03.2022

Принята 28.03.2022

Received 15.03.2022

Revised 25.03.2022

Accepted 28.03.2022