**ТЕХНИЧЕСКОЕ ЗАДАНИЕ   
Техническое задание на разработку интеллектуальной системы распознавания шахматных фигур с использованием нейронных сетей**

**1. Общие сведения**

**1.1. Наименование системы : Интеллектуальная система для распознавания шахматных фигур с использованием нейронных сетей.**

**1.2. Условное обозначение : ChessPieceRec (Chess Piece Recognition)**

**1.3. Основание для разработки : Настоящее техническое задание разработано в соответствии с требованиями ГОСТ 34.602-2020 и предназначено для регламентации процесса создания программного обеспечения, ориентированного на автоматизацию анализа изображений шахматных фигур с помощью методов глубокого обучения.**

**1.4. Разработчик : Частное образовательное учреждение высшего образования «Московский университет имени С.Ю. Витте»**

**1.5. Заказчик : Частное образовательное учреждение высшего образования «Московский университет имени С.Ю. Витте»**

**1.6. Цель разработки : Создание программного решения, предназначенного для автоматического распознавания шахматных фигур на изображениях, определения их цвета (белые/чёрные), а также предоставления пользователю результатов классификации через удобный графический интерфейс.**

**1.7. Источники разработки :**

* **Открытые датасеты с изображениями шахматных фигур.**
* **Научные исследования в области компьютерного зрения и сверточных нейронных сетей.**
* **Внутренние нормативные акты университета, касающиеся внедрения информационных систем.**
* **Технические средства и оборудование Центра дистанционных образовательных технологий.**

**1.8. Плановые сроки выполнения работ :**

* **Начало: 11.09.2024**
* **Окончание: 10.06.2025**

**1.9. Финансирование : Выполняется в рамках выпускной квалификационной работы бакалавра направления подготовки "Информационные системы и технологии".**

**2. Назначение и цели создания системы**

**2.1. Программная система предназначена для автоматизации процесса распознавания шахматных фигур на основе изображений реальных шахматных досок. Она позволяет:**

* **Идентифицировать тип фигуры (пешка, конь, слон, ладья, ферзь).**
* **Определять цвет фигуры (белая или чёрная).**
* **Предоставлять пользователю информацию о результатах анализа.**
* **Сохранять историю запросов и предоставлять статистику использования.**
* **Поддерживать возможность дообучения модели на новых данных.**

**2.2. Система может быть использована:**

* **В образовательных целях (в том числе в онлайн-курсах и тренингах).**
* **Для анализа шахматных партий.**
* **Как вспомогательный инструмент в тренировочных приложениях.**

**3. Требования к системе**

**3.1 Функциональные требования**

**3.1.1. Распознавание шахматных фигур**

* **Система должна поддерживать распознавание следующих шахматных фигур: пешка, конь, слон, ладья, ферзь, король.**
* **Должна быть реализована работа с изображениями, сделанными под различными углами и при разных условиях освещения.**
* **Поддержка форматов изображений: JPG, JPEG, PNG.**

**3.1.2. Определение цвета фигуры**

* **Система должна определять цвет фигуры (белая/чёрная) на основе анализа центральной части изображения.**
* **Модуль должен учитывать уровень яркости, контрастности и возможных оттенков.**

**3.1.3. Обработка изображений**

* **Система должна автоматически обрабатывать загружаемое изображение: изменять размер, нормализовать, устранять шумы.**
* **Должны применяться современные методы компьютерного зрения: OpenCV, PIL, ImageDataGenerator.**
* **Архитектура модели основана на сверточных нейронных сетях (CNN).**

**3.1.4. Интерфейс пользователя**

* **Интерфейс должен быть реализован в виде десктоп-приложения или веб-интерфейса.**
* **Должны присутствовать кнопки управления: «Загрузить изображение», «Начать сканирование», «Экспортировать результат».**
* **Интерфейс должен позволять просматривать историю запросов и статистику использования.**
* **Должна быть реализована функция drag-and-drop для загрузки изображений.**

**3.1.5. Минимизация ошибок распознавания**

* **Точность распознавания должна составлять не менее 95% на тестовой выборке.**
* **Уровень ошибок (False Positive / False Negative) не должен превышать 5%.**

**3.1.6. Возможность обучения модели**

* **Система должна предусматривать возможность дообучения модели на новых данных.**
* **Пользователь должен иметь возможность добавлять новые изображения и обновлять модель.**

**3.1.7. Генерация отчётов**

* **После каждой классификации система должна сохранять данные в лог-файл.**
* **Лог должен содержать: имя файла, предсказанный класс, цвет, уровень уверенности, дату и время.**
* **Должна быть предусмотрена визуализация статистики использования системы.**

**3.2 Нефункциональные требования**

**3.2.1. Производительность**

* **Время обработки одного изображения должно составлять не более 500 мс.**
* **Скорость работы модели должна соответствовать уровню производительности среднего персонального компьютера (Intel i5, 16 ГБ RAM, Windows/Linux).**

**3.2.2. Масштабируемость**

* **Система должна допускать расширение функционала для работы с видео и потоковой обработкой изображений.**
* **Возможность интеграции с мобильными устройствами и облачными сервисами.**

**3.2.3. Безопасность**

* **Система должна хранить данные локально, без передачи на внешние сервера.**
* **Должен быть предусмотрен контроль доступа к данным истории и настройкам модели.**

**3.2.4. Надежность**

* **Приложение должно корректно обрабатывать некорректные входные данные (например, файлы неверного формата, поврежденные изображения).**
* **Должны быть предусмотрены механизмы логирования ошибок и аварийного завершения.**

**3.2.5. Удобство использования**

* **Интерфейс должен быть простым и понятным для пользователей всех уровней подготовки.**
* **Должны быть предусмотрены подсказки и вспомогательные элементы интерфейса.**
* **Поддержка тем оформления (светлая/темная).**

**3.3 Требования к программному и аппаратному обеспечению**

**3.3.1. Поддерживаемые операционные системы :**

* **Microsoft Windows 10/11**
* **Linux (Ubuntu, Fedora)**
* **macOS (по возможности)**

**3.3.2. Минимальные аппаратные требования :**

* **Процессор: Intel Core i5 или аналог, не менее 4 ядер, тактовая частота не ниже 3.0 ГГц**
* **Оперативная память: не менее 8 ГБ**
* **Свободное место на диске: не менее 10 ГБ**
* **Графический процессор: желательно наличие, но не обязательно**

**3.3.3. Используемые технологии :**

* **Язык программирования: Python 3.10+**
* **Библиотеки машинного обучения: TensorFlow, Keras**
* **Библиотеки обработки изображений: OpenCV, PIL, ImageDataGenerator**
* **GUI: CustomTkinter (или Flask + HTML/CSS/JS для веб-версии)**
* **Хранение данных: CSV-файлы**

**4. Этапы разработки**

**4.1. Этап 1: Анализ требований**

* **Сбор информации от заказчика.**
* **Формирование технического задания и согласование его с заинтересованными сторонами.**

**4.2. Этап 2: Проектирование системы**

* **Разработка архитектуры модели глубокого обучения.**
* **Создание прототипа пользовательского интерфейса.**
* **Построение Use Case диаграммы и диаграммы последовательности.**

**4.3. Этап 3: Разработка программного обеспечения**

* **Реализация модели машинного обучения на базе CNN.**
* **Разработка графического интерфейса.**
* **Интеграция модулей: загрузка изображения → предобработка → классификация → вывод результата.**

**4.4. Этап 4: Тестирование системы**

* **Проверка точности распознавания.**
* **Анализ устойчивости к различным условиям освещения и ракурсам.**
* **Тестирование работы графического интерфейса и взаимодействия с пользователем.**

**4.5. Этап 5: Внедрение и интеграция**

* **Передача системы в Центр дистанционных образовательных технологий.**
* **Проведение демонстрации работы системы преподавательскому составу.**
* **Обучение пользователей работе с системой.**

**4.6. Этап 6: Поддержка и обновление**

* **Регулярное обновление модели на новых данных.**
* **Устранение найденных ошибок.**
* **Поддержка пользователей и сбор обратной связи для доработки.**

**5. Требования к документированию**

**5.1. Документация должна включать:**

* **Руководство пользователя.**
* **Руководство администратора.**
* **Техническое описание архитектуры системы.**
* **Описание используемых алгоритмов и модели глубокого обучения.**
* **Результаты тестирования и сравнительный анализ с аналогами.**
* **Примеры использования и рекомендации по эксплуатации.**

**5.2. Все документы должны быть оформлены в соответствии с установленными стандартами и сопровождать программную систему при внедрении.**

**6. Порядок контроля и приёмки**

**6.1. Контроль и приёмка системы проводятся в следующем порядке:**

* **Тестирование всех функциональных модулей программы.**
* **Проверка соответствия программного обеспечения требованиям, указанным в настоящем ТЗ.**
* **Оценка точности и скорости работы модели на тестовых и реальных данных.**
* **Подписание акта приемки разработанного программного обеспечения представителем заказчика.**

**7. Приложения**

**7.1. Примеры сценариев распознавания фигур:**

* **Распознавание фигуры на фотографии шахматной доски.**
* **Автоматическое определение типа и цвета.**
* **Отображение уровня уверенности модели.**

**7.2. Пример данных для обучения модели:**

* **Датасет содержит изображения шахматных фигур в различных условиях.**
* **Каждый класс (пешка, конь и т.д.) представлен не менее чем 1000 изображениями.**
* **Данные разделены на обучающую и валидационную выборки в соотношении 80/20.**