

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ястребов Олег Александрович

Должность: Ректор

Дата подписания: 25.05.2026 12:25:52

Уникальный программный ключ:

ca953a0120d891083f939673078ef1a989dae18a

Федеральное государственное автономное образовательное учреждение высшего образования

«Российский университет дружбы народов имени Патриса Лумумбы»

Факультет искусственного интеллекта

(наименование основного учебного подразделения (ОУП)-разработчика ОП ВО)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА СИСТЕМ КОМПЬЮТЕРНОГО ЗРЕНИЯ

(наименование дисциплины/модуля)

Рекомендована МССН для направлений подготовки:

**02.03.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ;**

09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

(код и наименование направления подготовки/специальности)

Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: РАЗРАБОТКА И ОБУЧЕНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

(наименование (профиль/специализация) ОП ВО)

2026 г.

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Проектирование и разработка систем компьютерного зрения» входит в программу бакалавриата «Искусственный интеллект: разработка и обучение интеллектуальных систем» по направлениям подготовки 02.03.02 Фундаментальная информатика и информационные технологии и 09.03.03 Прикладная информатика, и изучается в 6, 7, 8 семестрах 3, 4 курсов. Дисциплину реализует Кафедра прикладного искусственного интеллекта. Дисциплина состоит из 9 разделов и 81 тема и направлена на изучение полного цикла проектирования, разработки и эксплуатации промышленных систем компьютерного зрения: анализа требований заказчика и обследования предметной области, проектирования архитектуры CV-системы (выбор моделей, инфраструктуры, пайплайна обработки), организации сбора и разметки данных, итеративной разработки и обучения моделей, тестирования и валидации (функциональное, нагрузочное, приёмочное), оптимизации и развёртывания (контейнеризация, CI/CD, мониторинг), а также практик командной работы (Agile/Scrum, Git workflow, код-ревью, документирование) в контексте многосеместрового проектного цикла.

Целью освоения дисциплины является формирование у студентов навыков командной проектной работы над промышленной CV-системой в течение трёх семестров: от анализа требований и проектирования архитектуры до развёртывания, мониторинга и передачи заказчику, включая способность декомпозировать задачу, распределять роли в команде, управлять backlog-ом и спринтами, обеспечивать качество кода и данных, формулировать техническую документацию, проводить приёмочное тестирование, а также рефлексировать над процессом и результатами проекта.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Проектирование и разработка систем компьютерного зрения» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.2 Умеет анализировать альтернативные варианты решений для достижения намеченных результатов; разрабатывать план, определять целевые этапы и основные направления работ; УК-2.3 Владеет методиками разработки цели и задач проекта; методами оценки продолжительности и стоимости проекта, а также потребности в ресурсах;
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	УК-3.1 Знает типологию и факторы формирования команд, способы социального взаимодействия; УК-3.2 Умеет действовать в духе сотрудничества; принимать решения с соблюдением этических принципов их реализации; проявлять уважение к мнению и культуре других; определять цели и работать в направлении личностного, образовательного и профессионального роста; УК-3.3 Владеет навыками распределения ролей в условиях командного взаимодействия; методами оценки своих действий, планирования и управления временем;
УК-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе	УК-6.2 Умеет демонстрировать умение самоконтроля и рефлексии, позволяющие самостоятельно корректировать обучение по выбранной траектории;

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
	принципов образования в течение всей жизни	
ПК-1	Способен анализировать требования к программному обеспечению систем ИИ, разрабатывать технические спецификации и техническое задание на систему	ПК-1.3 Разрабатывает ТЗ на систему с элементами ИИ, проводит обследование текущей ситуации и выявление требований;
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	ПК-2.1 Проектирует архитектуру ИС с компонентами ИИ, выбирает архитектурные паттерны и технологический стек;
ПК-3	Способен разрабатывать и реализовывать стратегии тестирования и контроля качества программного обеспечения систем ИИ	ПК-3.2 Разрабатывает план тестирования и организационные документы для тестирования ПО систем ИИ;
DL-1	Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	DL-1.4 Разрабатывает и оптимизирует специализированные архитектуры для работы с изображениями, учитывая их уникальные свойства;
DL-3	Способен применять и (или) разрабатывать алгоритмы, методы и технологии компьютерного зрения	DL-3.1 Применяет (проводя выбор и эксперименты) известные алгоритмы и библиотеки компьютерного зрения, предобученные глубокие нейросетевые модели для прикладных задач анализа изображений и видеопотока, при необходимости дообучая и валидируя на собственных наборах данных; DL-3.2 Определяет стек технологий, методов и алгоритмов для построения продуктов с компьютерным зрением (системы видеоналитики, поисковые системы по изображениям и т.д.); DL-3.3 Имплементирует известные алгоритмы, архитектуры и модели компьютерного зрения на реальных данных, строит пайплайны обучения моделей и развертывания сервисов компьютерного зрения в продуктивной среде;
FC-2	Способен проводить передовые исследования в области фундаментальных и генеративных моделей	FC-2.2 Исследует и разрабатывает диффузионные и другие модели для несимвольных данных;
SS-1	Способен учитывать философские, когнитивные и социальные основания концепций ИИ в профессиональной деятельности	SS-1.2 Применяет методики работы с этическими и социальными рисками, возникающими на разных стадиях жизненного цикла ИИ;
SS-2	Способен к эффективной коммуникации и командной работе в междисциплинарных проектах в области ИИ	SS-2.1 Эффективно коммуницирует с участниками проектной команды при планировании, реализации и анализе результатов работы в контексте гибридной команды "Человек+ИИ", включая постановку задач людям и ИИ-агентам, фиксацию договорённостей и критериев качества; SS-2.2 Учитывает профессиональные и ролевые особенности коллег и контур ИИ-компонентов: адаптирует язык под аудиторию (tech/product/C-level), распределяет ответственность (RACI) и представляет результаты в понятном формате;
SS-3	Способен к критическому анализу, метарефлексии и переносу знаний при работе с системами ИИ	SS-3.2 Определяет релевантность применения ИИ для решения конкретных задач, анализирует поведение ИИ в техническом, социальном и правовом контекстах, переносит идеи и методы за пределы исходной предметной области;

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Проектирование и разработка систем компьютерного зрения» относится к блоку по выбору блока образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Проектирование и разработка систем компьютерного зрения».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
УК-3	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	Философия; Искусственный интеллект и когнитивная психология; Этика и безопасность использования искусственного интеллекта;	
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	Эксплуатационная практика (учебная); Правоведение; Этика и безопасность использования искусственного интеллекта; Методы разработки решений на основе искусственного интеллекта (Git, Docker);	
УК-6	Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни	<i>Основы программирования HTML - CSS - JavaScript**;</i> <i>Основы программирования на языке NodeJS**;</i> <i>Основы программирования на языке Go**;</i> <i>Основы программирования на языке Julia**;</i>	
ПК-1	Способен анализировать требования к программному обеспечению систем ИИ, разрабатывать технические спецификации и техническое задание на систему	<i>Правоведение;</i> <i>Параллельное и распределенное программирование;</i> <i>Введение в искусственный интеллект;</i> <i>Искусственный интеллект и когнитивная психология;</i> <i>Этика и безопасность использования искусственного интеллекта;</i> <i>Методы машинного обучения;</i> <i>История и теория программирования;</i> <i>Программирование на языке C++;</i> <i>Методы разработки решений на основе искусственного интеллекта (Git, Docker);</i> <i>Введение в базы данных;</i> <i>Онтология и графы знаний;</i> <i>Эксплуатационная практика</i>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
		<i>(учебная); Технологическая (проектно-технологическая) практика (учебная);</i>	
ПК-2	Способен проектировать архитектуру информационных систем с компонентами ИИ, разрабатывать прототипы и базы данных таких систем	<i>Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная); Программирование на языке C++; Параллельное и распределенное программирование; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Алгоритмы и структуры данных; Hadoop, SPARK; Программирование на языке Python; Введение в базы данных; Онтология и графы знаний;</i>	
ПК-3	Способен разрабатывать и реализовывать стратегии тестирования и контроля качества программного обеспечения систем ИИ	<i>Теория вероятностей и математическая статистика; Этика и безопасность использования искусственного интеллекта; Статистические методы и первичный анализ данных; Методы машинного обучения; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта; Анализ естественного языка с помощью методов искусственного интеллекта; Программирование на языке Python; Методы разработки решений на основе искусственного интеллекта (Git, Docker); Технологическая (проектно-технологическая) практика (учебная); Эксплуатационная практика (учебная);</i>	
SS-1	Способен учитывать философские, когнитивные и социальные основания концепций ИИ в профессиональной деятельности	<i>Философия; История и теория программирования; Введение в искусственный интеллект; Искусственный интеллект и когнитивная психология; Онтология и графы знаний; Методы машинного обучения; Лингвистические основы анализа естественного языка; Этика и безопасность использования искусственного</i>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
		<i>интеллекта;</i>	
SS-2	Способен к эффективной коммуникации и командной работе в междисциплинарных проектах в области ИИ	<i>Программирование на языке Python;</i> <i>Методы разработки решений на основе искусственного интеллекта (Git, Docker);</i> <i>Иностранный язык**;</i> <i>Русский язык (как иностранный)**;</i> <i>Иностранный язык в профессиональной деятельности**;</i> <i>Русский язык (как иностранный) в профессиональной деятельности**;</i> <i>Технологическая (проектно-технологическая) практика (учебная);</i> <i>Эксплуатационная практика (учебная);</i>	
SS-3	Способен к критическому анализу, метарефлексии и переносу знаний при работе с системами ИИ	<i>Эксплуатационная практика (учебная);</i> <i>Теория вероятностей и математическая статистика;</i> <i>Искусственный интеллект и когнитивная психология;</i> <i>Этика и безопасность использования искусственного интеллекта;</i> <i>Статистические методы и первичный анализ данных;</i> <i>Методы машинного обучения;</i> <i>Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i> <i>Анализ естественного языка с помощью методов искусственного интеллекта;</i> <i>Правоведение;</i> <i>Введение в искусственный интеллект;</i> <i>Введение в компьютерное зрение;</i> <i>Основы программирования HTML - CSS - JavaScript**;</i> <i>Основы программирования на языке NodeJS**;</i> <i>Основы программирования на языке Go**;</i> <i>Основы программирования на языке Julia**;</i> <i>Философия;</i>	
DL-1	Способен применять и (или) разрабатывать архитектуры глубоких нейронных сетей	<i>Анализ естественного языка с помощью методов искусственного интеллекта;</i> <i>Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i>	

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
DL-3	Способен применять и (или) разрабатывать алгоритмы, методы и технологии компьютерного зрения	<i>Введение в компьютерное зрение; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i>	
FC-2	Способен проводить передовые исследования в области фундаментальных и генеративных моделей	<i>Онтология и графы знаний; Лингвистические основы анализа естественного языка; Анализ естественного языка с помощью методов искусственного интеллекта; Введение в компьютерное зрение; Обработка и анализ изображений и видео с помощью методов искусственного интеллекта;</i>	

* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

** - элективные дисциплины /практики

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Проектирование и разработка систем компьютерного зрения» составляет «11» зачетных единиц.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.		Семестр(-ы)		
			6	7	8
<i>Контактная работа, ак.ч.</i>	162		68	52	42
Лекции (ЛК)	0		0	0	0
Лабораторные работы (ЛР)	0		0	0	0
Практически/семинарские занятия (СЗ)	162		68	52	42
<i>Самостоятельная работа обучающихся, ак.ч.</i>	153		85	65	3
<i>Контроль (экзамен/зачет с оценкой), ак.ч.</i>	81		27	27	27
Общая трудоемкость дисциплины	ак.ч.	396	180	144	72
	зач.ед.	11	5	4	2

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
Раздел 1	Инициация проекта: анализ требований, данные, архитектура	1.1	Введение в проектную разработку CV-систем	Дисциплина как проектный цикл: 3 семестра, командная работа, реальная задача. Жизненный цикл CV-проекта: постановка → данные → моделирование → тестирование → развёртывание → мониторинг. Формирование команд (3–5 человек). Распределение ролей: Team Lead, ML Engineer, Data Engineer, QA/DevOps. Agile/Scrum: спринты, daily standup, ретроспективы. Инструменты: GitHub Projects, Issues, Kanban	СЗ	УК-3.1, УК-3.2, SS-2.1
		1.2	Выбор задачи и анализ предметной области	Предложенные домены: промышленная инспекция дефектов, медицинская визуализация, ритейл (полки/товары), безопасность (видеонаблюдение), сельское хозяйство (дроны), спорт, экология. Команды выбирают задачу. Обследование предметной области: интервью с «заказчиком» (преподаватель), анализ аналогов, определение стейкхолдеров. Документирование: описание предметной области	СЗ	ПК-1.3, SS-1.2, УК-2.2
		1.3	Формулирование требований и анализ осуществимости	Функциональные требования: что система должна делать (детектировать, классифицировать, сегментировать). Нефункциональные: latency, throughput, accuracy, доступность, масштабируемость. Анализ осуществимости: достаточно ли данных, существуют ли предобученные модели, какие ресурсы нужны. User Stories. Критерии приёмки (Definition of Done). Документ: ТЗ (техническое задание) v1.0	СЗ	ПК-1.3, УК-2.3, ПК-3.2
		1.4	Стратегия сбора и разметки данных	Источники данных: открытые датасеты, веб-скрейпинг, собственная съёмка, синтетические данные. Оценка объёма: сколько данных нужно для каждой задачи (эмпирические правила). Стратегия разметки: инструменты (CVAT, LabelStudio, Roboflow), формат аннотаций, инструкции для разметчиков, контроль качества разметки (inter-annotator agreement). Документ: Data Collection Plan	СЗ	DL-3.2, SS-2.1
		1.5	Сбор и разметка данных — итерация 1	Практическая работа: команды собирают и размечают	СЗ	DL-3.2,

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			первую порцию данных (100–300 изображений). Распределение задач через GitHub Issues. Проверка качества: перекрёстная валидация разметки. Анализ: достаточно ли разнообразие, есть ли дисбаланс классов. EDA по собранным данным. Обновление Data Collection Plan		SS-2.1, УК-3.3
		1.6 Проектирование архитектуры CV-системы	Компоненты: источник данных → предобработка → модель → постобработка → API → хранилище результатов → мониторинг. Выбор модели: обоснование (YOLO vs. Faster R-CNN, U-Net vs. DeepLab, ResNet vs. ViT). Выбор фреймворка. Инфраструктура: Docker, GPU. Архитектурная диаграмма (C4 model, упрощённый). Документ: Architecture Decision Records (ADR)	СЗ	ПК-2.1, DL-3.1, DL-1.4
		1.7 Организация репозитория и CI/CD	Структура репозитория: data/, src/, configs/, models/, tests/, docs/, notebooks/. Git workflow: feature branches, PR, код-ревью. .gitignore, .dockerignore. DVC для данных. CI: GitHub Actions (lint, тесты, проверка формата). Pre-commit hooks. README.md: описание проекта, установка, использование. Практика: настройка инфраструктуры проекта	СЗ	SS-2.1, ПК-3.2, SS-2.2
		1.8 Baseline модель — быстрый прототип	Обучение baseline: простейшая модель на первичных данных (предобученная YOLO / ResNet с минимальным fine-tuning). Оценка: целевые метрики на валидации. Фиксация baseline метрик как «точка отсчёта». Анализ: на каких примерах ошибается, какие данные нужны. Практика: обучение, оценка, документирование baseline	СЗ	DL-3.2, DL-3.1
		1.9 Планирование спринтов и Definition of Done	Декомпозиция оставшейся работы на спринты (2 недели). Backlog: User Stories → задачи (Issues) → оценка (story points). Kanban-доска в GitHub Projects. Definition of Done для ML-задач: код, тесты, код-ревью, документация, метрики. Ретроспектива первых недель: что получилось, что улучшить	СЗ	УК-2.2, УК-3.2, SS-2.1
		1.10 Итерация 2: расширение данных и улучшение модели	Сбор дополнительных данных по выявленным пробелам. Аугментация: Albumentations pipeline. Улучшение модели: подбор гиперпараметров, другая архитектура, transfer learning. Логирование экспериментов (W&B / MLflow).	СЗ	DL-3.2, DL-3.1, SS-2.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				Сравнение с baseline. Код-ревью PR внутри команды		
		1.11	Промежуточная презентация: результаты инициации	Подготовка промежуточного отчёта: описание задачи, данных, архитектуры, baseline, текущие результаты. Формат: 10-минутная презентация + демонстрация. Обратная связь от преподавателя и других команд. Фиксация action items	СЗ	УК-3.3, SS-3.2
		1.12	Ретроспектива и планирование следующей фазы	Ретроспектива команды: что сработало, что нет, какие процессы улучшить. Обновление ТЗ по итогам feedback. Планирование: приоритеты на фазу разработки (данные, модель, тестирование). Рефлексия индивидуального вклада каждого участника. Обновление README и документации	СЗ	УК-6.2, УК-3.2, SS-1.2
Раздел 2	Разработка: итеративное улучшение модели и данных	2.1	Итерация 3: целенаправленное улучшение данных	Анализ ошибок модели: на каких категориях/условиях модель слабая. Целенаправленный сбор данных для слабых мест (hard examples). Active learning (обзор): приоритизация разметки. Очистка данных: удаление некачественной разметки, дубликатов. Практика: error-driven data collection	СЗ	DL-3.2, DL-3.3, SS-3.2
		2.2	Итерация 4: продвинутая аугментация и предобработка	Mosaic, CutMix для детекции. Domain-specific аугментации (шум, освещение, ракурс для конкретного домена). Crop-paste augmentation для instance segmentation. Test-Time Augmentation (ТТА). Практика: разработка и тестирование custom аугментации для проекта	СЗ	DL-3.2, DL-1.4
		2.3	Эксперименты с архитектурами	Сравнение нескольких архитектур для задачи проекта: accuracy vs. latency vs. model size. Ablation studies: влияние каждого компонента (backbone, neck, head). Ensemble моделей (обзор). Практика: таблица сравнения моделей, обоснование финального выбора	СЗ	DL-3.1, DL-1.4, FC-2.2
		2.4	Оптимизация гиперпараметров и стратегии обучения	Learning rate scheduling: warmup + cosine decay. Optimizer: AdamW vs. SGD+momentum. Batch size и effective batch (gradient accumulation). Mixed precision training. Подбор гиперпараметров: Optuna. Практика: систематический подбор, логирование в W&B, визуализация	СЗ	DL-3.2, DL-3.1
		2.5	Постобработка и бизнес-логика	Постобработка для детекции: NMS, confidence filtering, класс-специфичные пороги. Постобработка для сегментации: CRF, морфологические операции, удаление мелких компонент. Бизнес-логика: подсчёт объектов,	СЗ	DL-3.3, ПК-2.1

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			классификация сцены, генерация отчёта. Практика: реализация постобработки для проекта		
		2.6 Тестирование ML-моделей: стратегия и план	Уровни тестирования CV-системы: unit (предобработка, постобработка), integration (пайплайн end-to-end), model (метрики на test set), system (API, latency). Тестирование по подгруппам: мелкие/крупные объекты, разное освещение, разный фон. Edge cases: пустые изображения, повреждённые файлы, необычные разрешения. Документ: Test Plan	СЗ	ПК-3.2, DL-1.4
		2.7 Реализация тестов	Написание тестов: pytest для предобработки и постобработки, тесты пайплайна (вход → ожидаемый формат выхода), тесты метрик (регрессионные: accuracy ≥ порог на фиксированном test set). Интеграция в CI: GitHub Actions запускает тесты при PR. Практика: покрытие ключевых компонентов тестами	СЗ	ПК-3.2, SS-2.2
		2.8 Код-ревью и качество кода	Процесс код-ревью: чек-лист (корректность, читаемость, тесты, документация, отсутствие утечки данных). Рефакторинг: выделение переиспользуемых компонентов. Линтинг: black, flake8, isort. Type hints. Docstrings. Практика: перекрёстное ревью между командами (inter-team review)	СЗ	SS-2.2, SS-2.1
		2.9 Итерация 5: финальное улучшение модели в семестре	Применение всех наработок: расширенные данные, оптимальные гиперпараметры, лучшая архитектура, постобработка. Финальная оценка на held-out test set. Сравнение с baseline: таблица прогресса по итерациям. Замораживание модели v1.0	СЗ	DL-3.2, DL-3.1
		2.10 Промежуточный отчёт: прогресс по модели	Обновление документации: текущие метрики, сравнение с baseline, описание экспериментов. Демонстрация: визуализация предсказаний, примеры ошибок. Обратная связь от преподавателя. Планирование развёртывания	СЗ	УК-3.3, SS-3.2
		2.11 Ретроспектива и планирование развёртывания	Ретроспектива итераций 3–5: что дало наибольший прирост качества. Обновление backlog: задачи на развёртывание, тестирование, документирование. Распределение ответственности. Оценка технического долга: что нужно отрефакторить перед production	СЗ	УК-6.2, УК-2.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
Раздел 3	Развёртывание, мониторинг и сдача MVP	3.1	Контейнеризация CV-сервиса	Dockerfile для инференс-сервиса: базовый образ с CUDA, установка зависимостей, копирование модели и кода. Docker Compose: сервис инференса + Redis (кэш) + PostgreSQL (результаты). Health check. .dockerignore. Практика: контейнеризация пайплайна проекта	СЗ	ПК-2.1, DL-3.3
		3.2	API инференса: проектирование и реализация	FastAPI: эндпоинты /predict (single image), /predict_batch, /health. Pydantic-модели для запросов и ответов. Обработка ошибок: невалидный формат, таймауты. Swagger-документация. Async обработка (при необходимости). Практика: реализация API для проекта	СЗ	ПК-2.1, DL-3.3
		3.3	Оптимизация инференса	Экспорт: TorchScript, ONNX. Оптимизация: ONNX Runtime, TensorRT (при наличии GPU). Замеры: latency (p50, p95, p99), throughput (images/sec). Батчирование запросов. Практика: оптимизация модели проекта, таблица «формат → latency → accuracy»	СЗ	DL-3.3, FC-2.2
		3.4	CI/CD для CV-проекта	CI: lint → unit tests → model regression tests → build Docker image. CD: автоматическая публикация образа в registry при merge в main. Тегирование версий. Практика: настройка полного CI/CD pipeline в GitHub Actions	СЗ	ПК-3.2, SS-2.1
		3.5	Мониторинг и логирование	Логирование запросов и предсказаний (structured logging). Мониторинг: latency, throughput, error rate (Prometheus + Grafana, обзор). Мониторинг дрейфа: распределение confidence, распределение предсказанных классов. Алерты при отклонениях. Практика: настройка логирования и базового мониторинга	СЗ	DL-3.3, ПК-3.2
		3.6	Нагрузочное тестирование	Инструменты: locust, wrk. Сценарии: single request latency, concurrent requests, sustained load. Определение пропускной способности сервиса. Bottleneck analysis: CPU vs. GPU vs. network vs. preprocessing. Практика: нагрузочное тестирование API проекта, отчёт с рекомендациями	СЗ	ПК-3.2, DL-3.3
		3.7	Техническая документация проекта v1.0	Документы: ТЗ (обновлённая версия), Architecture Decision Records, Model Card, Datasheet for Dataset, API спецификация (OpenAPI), план тестирования, инструкция по развёртыванию. README: установка, запуск, использование. Практика: финализация документации для	СЗ	ПК-1.3, SS-1.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				MVP		
		3.8	Приёмочное тестирование MVP	Проведение приёмочного тестирования по критериям из ТЗ: метрики \geq пороги, latency \leq SLA, edge cases, API корректность. Протокол тестирования: результаты, несоответствия, рекомендации. Практика: перекрёстное приёмочное тестирование (одна команда тестирует другую)	СЗ	ПК-3.2, DL-1.4
		3.9	Демонстрация MVP заказчику	Презентация MVP: демонстрация работающей системы, ключевые метрики, архитектура, ограничения. Формат: 15 мин + вопросы. Обратная связь от «заказчика» (преподаватель). Фиксация замечаний и пожеланий к доработке	СЗ	УК-3.3, SS-3.2
		3.10	Ретроспектива семестра 6	Ретроспектива команды: процессы, техническое качество, коммуникация. Анализ скорости (velocity): запланировано vs. выполнено. Технический долг: что осталось. Индивидуальная рефлексия: чему научился, что хочу улучшить	СЗ	УК-6.2, УК-3.2
		3.11	Планирование семестра 7	Анализ feedback от «заказчика» и перекрёстного тестирования. Приоритизация: критичные доработки, улучшения, nice-to-have. Обновление backlog. Перераспределение ролей (при необходимости). Установка milestones для семестра 7	СЗ	УК-2.2, УК-2.3
Раздел 4	Доработка и расширение функциональности	4.1	Анализ feedback и планирование доработок	Детальный разбор замечаний по MVP: недостаточная точность на подклассах, проблемы с edge cases, UX API. Декомпозиция доработок на задачи. Sprint planning на семестр. Обновление ТЗ v2.0 с учётом новых требований	СЗ	УК-2.3, ПК-1.3
		4.2	Расширение датасета и переобучение	Целенаправленный сбор данных по слабым местам, выявленным в MVP. Применение pseudo-labeling для расширения (обзор). Переобучение модели на расширенном датасете. Сравнение метрик: до и после. Обновление Datasheet	СЗ	DL-3.2, DL-3.1
		4.3	Дополнительная функциональность: трекинг и видеоанализ	Реализация дополнительных фиच по запросу «заказчика»: трекинг объектов (ByteTrack / DeepSORT), подсчёт объектов по зонам, классификация событий на видео. Практика: добавление нового функционала, покрытие тестами, код-ревью	СЗ	DL-3.3, DL-3.1

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
		4.4	Дополнительная функциональность: отчёты и визуализация	Dashboard для отображения результатов: Streamlit / Gradio. Генерация отчётов: PDF / JSON с агрегированными результатами. Визуализация предсказаний: наложение масок, bbox, heatmaps. Практика: реализация интерфейса для проекта	СЗ	DL-3.3, ПК-2.1
		4.5	Edge deployment и оптимизация для ограниченных ресурсов	Развёртывание на edge-устройствах: NVIDIA Jetson, мобильные (обзор). Ограничения: память, вычисления, энергопотребление. Оптимизация: quantization (INT8), pruning, knowledge distillation. ONNX Runtime Mobile. Практика: экспорт оптимизированной модели для edge (если применимо к проекту)	СЗ	FC-2.2, DL-3.3
		4.6	Обработка видеопотока в реальном времени	Архитектура: захват кадров → предобработка → инференс → постобработка → визуализация. Буферизация и пропуск кадров при перегрузке. Асинхронная обработка: отдельные потоки для захвата и инференса. Замер FPS. Практика: обработка видеопотока для проекта (файл или webcam)	СЗ	DL-3.3, DL-3.1
		4.7	Масштабирование и отказоустойчивость	Масштабирование: несколько экземпляров за load balancer (Nginx, обзор). Kubernetes: pod, service, deployment (обзор). Отказоустойчивость: health checks, restart policies, graceful degradation. Практика: Docker Compose с несколькими репликами сервиса	СЗ	ПК-2.1, DL-3.3
		4.8	Интеграционное тестирование доработок	Полное интеграционное тестирование: API → модель → постобработка → хранилище → визуализация. Тестирование на реальных сценариях использования. Тестирование recovery после сбоев. Обновление Test Plan и результатов тестирования	СЗ	ПК-3.2, SS-2.2
		4.9	Промежуточная демонстрация: прогресс доработок	Демонстрация улучшений: новые метрики, новая функциональность, результаты тестирования. Сравнение с MVP. Обратная связь от преподавателя и других команд. Корректировка плана на оставшуюся часть семестра	СЗ	УК-3.3, SS-3.2
		Раздел 5	Финализация продукта и документации	5.1	Финальная оптимизация модели	Последние улучшения: подбор порогов под бизнес-задачу, TTA, Stochastic Weight Averaging (при необходимости). Экспорт финальной модели в production-формат. Фиксация версии: модель, данные, код. Тегирование release v2.0 в Git
5.2	Финальное приёмочное тестирование			Полное приёмочное тестирование по обновлённому ТЗ v2.0.	СЗ	ПК-3.2,

Номер раздела	Наименование раздела дисциплины	Наименование темы	Содержание темы	Вид учебной работы *	Формируемые индикаторы
			Регрессионные тесты: метрики не ухудшились относительно v1.0. Нагрузочное тестирование: повторные замеры. Тестирование edge cases. Протокол приёмки: PASS/FAIL по каждому критерию		DL-1.4
		5.3 Оценка справедливости и этических аспектов	Анализ модели по подгруппам (если применимо): разные условия съёмки, разные типы объектов, различные демографические группы. Идентификация потенциальных bias. Формулирование ограничений использования. Обновление Model Card: раздел «Ethical Considerations» и «Limitations»	СЗ	SS-3.2, SS-1.2
		5.4 Финализация технической документации	Ревизия всех документов: ТЗ v2.0, ADR, Model Card, Datasheet, API спецификация, Test Plan, инструкция по развёртыванию. Consistency check: документация соответствует текущему состоянию системы. Генерация API-документации (Swagger). Практика: финальная вычитка и обновление	СЗ	ПК-1.3, SS-1.2
		5.5 Подготовка delivery package	Формирование пакета передачи: Docker image (опубликован), исходный код (GitHub, tag release), документация, модель (DVC / Model Registry), данные (DVC), тесты, CI/CD pipeline. Инструкция по развёртыванию с нуля. Практика: проверка — новый человек может развернуть систему по инструкции	СЗ	ПК-2.1, SS-2.1
		5.6 Подготовка финальной презентации	Структура: бизнес-задача → требования → архитектура → данные → модель → результаты → демонстрация → ограничения → дальнейшие шаги. Демонстрация: live demo работающей системы. Распределение частей между членами команды. Репетиция	СЗ	УК-3.3, SS-3.2
		5.7 Индивидуальная рефлексия по итогам двух семестров	Каждый участник готовит индивидуальный отчёт: мой вклад в проект, чему научился, какие навыки развил, что сделал бы иначе. Анализ: как распределялась нагрузка, как менялась моя роль. Связь с планированием профессионального роста	СЗ	УК-6.2, SS-1.2
		5.8 Ретроспектива семестра 7	Командная ретроспектива: что улучшилось по сравнению с семестром 6, какие процессы теперь работают хорошо. Технический долг: финальная оценка. Velocity: сравнение	СЗ	УК-3.2, УК-6.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				семестров. Формулирование «памятки» для следующей команды		
		5.9	Планирование семестра 8: защита и peer review	Планирование финальной фазы: расписание защит, подготовка к перекрёстному ревью. Определение оставшихся доработок (минимальных). Распределение ответственности за подготовку финальных артефактов	СЗ	УК-2.2, УК-2.3
Раздел 6	Защита проекта и peer review	6.1	Защита проекта — команды 1 и 2	Презентация (20 мин) + live демонстрация (10 мин) + вопросы (10 мин). Критерии: качество модели, архитектура, код, тесты, документация, командная работа, обоснованность решений. Оценка преподавателем и peer review другими командами по чек-листу	СЗ	УК-3.3, DL-3.2
		6.2	Защита проекта — команды 3 и 4	Аналогичный формат защиты. Перекрёстное оценивание: каждая команда оценивает другую по единому чек-листу (метрики, документация, код, демо, презентация, ответы на вопросы)	СЗ	УК-3.3, DL-3.2
		6.3	Защита проекта — оставшиеся команды	Аналогичный формат. Обсуждение: общие паттерны (что получилось у всех, типичные проблемы, лучшие находки)	СЗ	УК-3.3, SS-3.2
		6.4	Перекрёстное ревью кода и системы	Команды обмениваются репозиториями. Код-ревью по чек-листу: структура, читаемость, тесты, документация, воспроизводимость. Попытка развёртывания чужого проекта по инструкции. Формирование отчёта ревью с рекомендациями	СЗ	SS-2.2, ПК-3.2
		6.5	Перекрёстное приёмочное тестирование	Команды проводят приёмочное тестирование чужого проекта по его ТЗ. Проверка: метрики, edge cases, API, документация. Формирование протокола приёмки. Обсуждение расхождений ожиданий и реальности	СЗ	ПК-3.2, DL-1.4
		6.6	Общая ретроспектива и lessons learned	Командные ретроспективы: ключевые уроки за 2 семестра. Обсуждение на уровне потока: общие проблемы и лучшие практики. Формулирование «памятки разработчика CV-систем». Связь с промышленной практикой	СЗ	УК-6.2, УК-3.2
		6.7	Анализ feedback по защите и планирование семестра 8	Разбор обратной связи от защиты и перекрёстного ревью. Приоритизация финальных доработок. Планирование семестра 8: исследовательское расширение, научная статья, финальная передача. Формулирование индивидуальных целей на семестр	СЗ	УК-2.3, SS-1.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
		6.8	Фиксация промежуточных результатов	Обновление всех артефактов проекта по итогам feedback: код, документация, модель. Тегирование v2.1 в Git. Подготовка backlog семестра 8. Формальное закрытие спринтов семестра 7	СЗ	SS-2.1, УК-2.2
Раздел 7	Исследовательское расширение и продвинутые техники	7.1	Исследование новых подходов для проекта	Обзор литературы: новейшие архитектуры и методы для задачи проекта (последние 1–2 года). Foundation models для CV: SAM, DINOv2, CLIP — применимость к задаче проекта. Формулирование гипотез: какой новый метод может улучшить систему. Планирование экспериментов	СЗ	FC-2.2, DL-3.1
		7.2	Эксперимент с foundation model	Применение foundation model (SAM / DINOv2 / CLIP) к задаче проекта: zero-shot или fine-tuning. Сравнение с текущей моделью: метрики, latency, требования к данным. Документирование результатов эксперимента. Обсуждение: когда foundation models оправданы, когда — нет	СЗ	FC-2.2, DL-3.1
		7.3	Self-supervised pretraining и domain adaptation	Self-supervised pretraining на неразмеченных данных домена (MAE, DINO — обзор). Domain adaptation: source → target domain (обзор методов). Практика: предобучение на неразмеченных данных проекта → дообучение на размеченных. Оценка прироста	СЗ	DL-3.2, FC-2.2
		7.4	Генерация синтетических данных	Подходы: рендеринг (Blender, обзор), GAN-based augmentation, диффузионные модели для генерации обучающих данных. Copy-paste augmentation. Практика: генерация синтетических данных для проекта, обучение с добавлением синтетики, оценка влияния	СЗ	DL-3.2, DL-3.3
		7.5	Объяснимость и trust в CV-системах	Grad-CAM, attention maps для анализа решений модели. Counterfactual explanations для CV (обзор). Uncertainty estimation: MC Dropout, ensemble disagreement. Отказ от предсказания при низкой уверенности. Практика: добавление модуля объяснимости в систему проекта	СЗ	DL-1.4, SS-3.2
		7.6	Мониторинг дрейфа данных в production	Мониторинг входных данных: распределение яркости, размеров объектов, соотношения классов. Обнаружение дрейфа: PSI, KL-divergence на эмбедах. Стратегия: алерт → ручная проверка → переобучение. Практика: реализация мониторинга дрейфа для проекта	СЗ	DL-3.3, ПК-3.2
		7.7	Интеграция исследовательских	Выбор лучших наработок: какие эксперименты дали	СЗ	DL-3.2,

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
			результатов в production	значимое улучшение. Интеграция в основную ветку: рефакторинг, тесты, код-ревью. Обновление модели в production (версионирование). Обновление документации и Model Card. Тегирование release v3.0		SS-2.1
Раздел 8	Финальная доработка и передача проекта	8.1	Финальная оптимизация и замораживание	Последний цикл оптимизации: финальная модель, оптимальные пороги, оптимизация инференса (ONNX/TensorRT). Замораживание: никаких изменений модели после этой точки. Фиксация всех версий: код, данные, модель, зависимости	СЗ	DL-3.3, DL-3.2
		8.2	Полное регрессионное и приёмочное тестирование	Финальный раунд тестирования: unit, integration, model regression, нагрузочные, edge cases, приёмочные. Все тесты зелёные. Протокол финального тестирования: подписание. Сравнение с ТЗ v2.0: все критерии выполнены	СЗ	ПК-3.2, DL-1.4
		8.3	Написание технического отчёта / научной статьи	Формат: технический отчёт (10–15 страниц) или краткая статья (4–6 страниц, формат конференции). Структура: введение, обзор методов, описание системы, эксперименты, результаты, заключение. Практика: написание черновика, взаимное ревью между командами	СЗ	SS-1.2, SS-3.2
		8.4	Финализация всей документации	Полная ревизия: ТЗ v2.0, ADR, Model Card v3.0, Datasheet, API spec, Test Plan, Test Report, инструкция по развёртыванию, технический отчёт, README. Consistency check. Публикация документации (GitHub Pages / Wiki). Практика: финальная вычитка всеми участниками	СЗ	ПК-1.3, SS-1.2
		8.5	Формирование финального delivery package	Пакет передачи: Docker image (финальный), исходный код (тег v3.0), документация, модель, данные, тесты, CI/CD, технический отчёт. Проверка: третье лицо (другая команда) может развернуть по инструкции. Формальная передача «заказчику»	СЗ	ПК-2.1, SS-2.1
		8.6	Подготовка финальной презентации и демонстрации	Структура: эволюция проекта за 3 семестра → финальная архитектура → ключевые результаты → исследовательские находки → live demo → ограничения → рекомендации по развитию. Распределение между участниками. Репетиция	СЗ	УК-3.3, SS-3.2
		8.7	Индивидуальная итоговая рефлексия	Итоговый индивидуальный отчёт: мой путь в проекте за 3 семестра, эволюция навыков, ключевые уроки, что сделал бы иначе, как проект повлиял на профессиональное	СЗ	УК-6.2, SS-1.2

Номер раздела	Наименование раздела дисциплины	Наименование темы		Содержание темы	Вид учебной работы *	Формируемые индикаторы
				самоопределение. Формат: 3–4 страницы. Связь с карьерными целями		
Раздел 9	Финальная защита и подведение итогов	9.1	Финальная защита — команды 1 и 2	Итоговая презентация (25 мин) + live demo (10 мин) + вопросы (15 мин). Критерии: эволюция за 3 семестра, качество финальной системы, качество кода и тестов, документация, командная работа, исследовательский вклад, обоснованность решений. Оценка преподавателем и внешним экспертом (при возможности)	СЗ	УК-3.3, DL-3.2
		9.2	Финальная защита — команды 3 и 4	Аналогичный формат. Перекрёстное оценивание по финальному чек-листу	СЗ	УК-3.3, DL-3.2
		9.3	Финальная защита — оставшиеся команды	Аналогичный формат. Общее обсуждение: сравнение подходов разных команд, лучшие практики	СЗ	УК-3.3, SS-3.2
		9.4	Финальное перекрёстное ревью	Команды обмениваются финальными пакетами. Последнее ревью: развёртывание по инструкции, проверка тестов, проверка документации. Формирование финального отчёта ревью. Обсуждение: насколько система готова к реальной эксплуатации	СЗ	SS-2.2, ПК-3.2
		9.5	Итоговая ретроспектива: 3 семестра проекта	Командная ретроспектива за весь проект: ключевые milestone-ы, кризисы и их преодоление, главные достижения. Обсуждение на уровне потока: общие уроки. Формулирование «книги знаний» (knowledge base) для будущих потоков	СЗ	УК-6.2, УК-3.2
		9.6	Карта профессионального развития	Обзор пройденного пути: от идеи до работающей промышленной CV-системы. Обсуждение: какие навыки наиболее востребованы в индустрии. Рекомендации: open source, Kaggle, стажировки, магистратура. Составление индивидуальных планов развития на ближайший год	СЗ	УК-6.2, SS-1.2
		9.7	Подведение итогов дисциплины	Формальное закрытие проектов. Публикация лучших проектов (GitHub, портфолио). Обратная связь по дисциплине: что улучшить. Благодарности. Связь с ВКР: возможность развития проекта в дипломную работу	СЗ	УК-3.1, SS-1.2

* - заполняется только по **ОЧНОЙ** форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Семинарская	Аудитория для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная комплектом специализированной мебели и техническими средствами мультимедиа презентаций.	Персональные компьютеры, необходимое ПО
Для самостоятельной работы	Аудитория для самостоятельной работы обучающихся (может использоваться для проведения семинарских занятий и консультаций), оснащенная комплектом специализированной мебели и компьютерами с доступом в ЭИОС.	Персональные компьютеры, необходимое ПО

* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература:

1. Сацюк, А. В. Компьютерное зрение. Практика : учебное пособие / А. В. Сацюк. — Москва, Вологда : Инфра-Инженерия, 2025. — 272 с. — ISBN 978-5-9729-2346-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/154568.html>

2. Шапиро, Л. Компьютерное зрение: учебник / Л. Шапиро, Д. Стокман. - 5-е изд. - Москва: Лаборатория знаний, 2024. - 762 с. - ISBN 978-5-93208-725-1. - Текст: электронный. - URL: <https://znanium.ru/catalog/product/2167351>

Дополнительная литература:

1. Цифровая обработка изображений в OpenCv. Практикум: учебное пособие для вузов / А. И. Матвеев. - 2-е изд., стер. - Санкт-Петербург: Лань, 2023. - 104 с. - ISBN 978-5-507-46249-0. - <https://e.lanbook.com/book/266783>

2. Афанасьева, Жанна Сергеевна. Распознавание объектов с помощью сверточных нейронных сетей: учебное пособие. - Иркутск: Издательство Иркутского национального исследовательского технического университета, 2023 (Иркутск). - 133 с.: ил.; ISBN 978-5-8038-1914-1

3. Селянкин, В. В. Компьютерное зрение. Анализ и обработка изображений: учебное пособие / В. В. Селянкин. — Санкт-Петербург: Лань, 2019. — 152 с. — ISBN 978-5-8114-3368-1. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/113938>

Ресурсы информационно-телекоммуникационной сети «Интернет»:

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров

- Электронно-библиотечная система РУДН – ЭБС РУДН
<https://mega.rudn.ru/MegaPro/Web>
- ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>
- ЭБС «Юрайт» <http://www.biblio-online.ru>
- ЭБС «Консультант студента» www.studentlibrary.ru
- ЭБС «Знаниум» <https://znanium.ru/>

2. Базы данных и поисковые системы

- Sage <https://journals.sagepub.com/>
- Springer Nature Link <https://link.springer.com/>
- Wiley Journal Database <https://onlinelibrary.wiley.com/>
- Научометрическая база данных Lens.org <https://www.lens.org>

Учебно-методические материалы для самостоятельной работы обучающихся при освоении дисциплины/модуля:*

* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**