

Аннотация рабочей программы дисциплины

Б1.В.05 «Технологии компьютерного зрения»

Направление

подготовки/специальность 01.03.02 Прикладная математика и информатика

(код и наименование направления подготовки/специальности)

Курс 1 Семестр 3 Количество з.е. 3

Объем трудоемкости: 3 зачетных единиц (108 ч., из них – 68 час. аудиторной нагрузки: лекционных 34 ч., лабораторных работ - 34 ч., 30,8 часов самостоятельной работы, 4 часов КСР, 0,2 часа ИКР, выполнение курсового проекта 5), форма контроля – зачет, курсовой проект

Цель дисциплины: Обеспечить системное изучение фундаментальных основ компьютерного зрения, подкрепленное интенсивной практикой на современной инструментальной базе, и создает прочную базу для дальнейшего изучения области, включая машинное обучение.

Задачи дисциплины:

1. Сформировать понимание изображения как математической модели и данных.
2. Научить решать базовые задачи КЗ: фильтрация, сегментация, детекция, трекинг, калибровка.
3. Развить навыки реализации алгоритмов КЗ на C++ с использованием OpenCV.
4. Дать представление о физике формирования изображения и характеристиках камер.
5. Подготовить базу для курсов по машинному обучению и глубокому обучению в КЗ.

Место дисциплины в структуре ООП ВО:

Дисциплина «Технологии компьютерного зрения» относится к «Часть, формируемая участниками образовательных отношений» Блока 1 «Дисциплины (модули)» учебного плана.

Дисциплина использует знания и умения студентов, полученные в рамках изучения дисциплин: Программирование, Алгоритмы и структуры данных, Алгебра и аналитическая геометрия, Математический анализ, Алгебра и введение в тензорный анализ.

Результаты освоения дисциплины будут использоваться при прохождении следующих дисциплин/практик: Операционные системы, учебная (технологическая (проектно-технологическая) практика), Физика, Нейросетевые технологии, Современные методы компьютерного зрения, ИИ в робототехнике.

Результаты обучения (знания, умения, опыт, компетенции):

Содержание и структура дисциплины:

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих

Код и наименование индикатора*	Результаты обучения по дисциплине
SS-2 Способен осуществлять свою трудовую деятельность с учётом необходимости эффективной коммуникации и взаимодействия в рамках коллективной проектной работы в сфере ИИ	
SS-2.2. Учитывает профессиональные и ролевые особенности коллег и контур ИИ-компонентов: адаптирует язык под	Участствует в подготовке презентации по своей части (например, рассказывает про архитектуру ИИ-системы или метрики). Принимает предложенное распределение

Код и наименование индикатора*	Результаты обучения по дисциплине
аудиторию (tech/product/C-level), распределяет ответственность (RACI) и представляет результаты в понятном формате	ролей без активного участия в общей координации. Ориентируется в структуре общего результата проекта. Указывает источники данных и ссылку на код/модель, демонстрирует понимание базовых ограничений работы с данными (приватность, лицензии)
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.	Использует готовые модели из библиотеки OpenCV и популярных фреймворков (TensorFlow Hub, TorchVision) Использует базовые функции библиотек OpenCV и PIL для обработки.
DL-3.2 Определяет стек технологий, методов и алгоритмов для построения продуктов с компьютерным зрением (системы видеоаналитики, поисковые системы по изображениям и т.д.)	Понимает принципы представления изображений и кодирования цвета; применяет фильтрацию изображений, включая частотную; владеет аппаратом математической морфологии. Понимает базовые задачи анализа изображений - классификация, детекция, сегментация. применяет преобразование Хафа и RANSAC; применяет алгоритмы детекции характеристических точек (детектор Харриса, детектор Фестнера, SUSAN, блябы, DoG); применяет дескрипторы изображений, например, SIFT.
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	Использует готовые реализации из библиотек Умеет оценивать качество по стандартным метрикам (точность, mAP, IoU).
DL-3.4 Разрабатывает новые алгоритмы и библиотеки компьютерного зрения, новые архитектуры глубоких нейронных сетей и методы их обучения для задач анализа изображений и видео	Владеет аппаратом эпполярной геометрии. Способен применять алгоритмы стереозрения. Знает математические основы представления поверхностей.
PL-4 Способен применять языки программирования C/C++ для решения задач в области ИИ	
PL-4.1 Разрабатывает и отлаживает эффективные многопоточные решения на C++, тестирует, испытывает и оценивает качество таких решений	Знает основы синтаксиса языка. Знает общие принципы параллельных вычислений и понимает проблемы, возникающие при распараллеливании алгоритмов.
PL-4.2 Разрабатывает и отлаживает системы ИИ на C++ под конкретные аппаратные платформы с ограничениями по вычислительной мощности, в том числе для встроенных систем	Знает основы синтаксиса языка, основы построения систем ИИ. Пишет небольшие программы для обработки данных. Понимает возможности и ограничения встроенных систем. Находит и использует библиотеки, соответствующие решаемой задаче
PL-4.3 Разрабатывает и отлаживает решения на C++, использующие GPU и FPGA для массовой параллелизации вычислений в рамках общей системы ИИ, как с применением готовых решений, так и с разработкой своих	Знает основы синтаксиса языка C/C++, основы построения систем ИИ, общие принципы параллельных вычислений Умеет использовать средства отладки и профилирования кода, находить участки кода, ограничивающие производительность системы.

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 3 семестре

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Основы OpenCV и цветовые пространства	17	6		6	5
2.	Фильтрация и морфология	16,8	6		6	5
3.	Поиск и анализ контуров	17	6		6	5
4.	Пространственные преобразования	11	4		4	4,8
5.	Детекция объектов и сопоставление	6	2		2	2
6.	Детекция движения и фон	6	2		2	2
7.	Трекинг объектов	6	2		2	2
8.	Калибровка камеры и устранение искажений	17	6		6	5
ИТОГО по разделам дисциплины		98,8	34		34	30,8
Контроль самостоятельной работы (КСР)		4				
Промежуточная аттестация (ИКР)		0,2				
Выполнение курсового проекта (КРП)		5				
Общая трудоемкость по дисциплине		108				

Примечание: Л – лекции, ПЗ – практические занятия/семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

Курсовые проекты или работы.

Обработка изображений

1. Модификация драйвера USB-камеры для применения простых аппаратных корректировок (яркость, контрастность) через интерфейс V4L2 в Astra Linux.
2. Добавление поддержки формата изображения MJPG в драйвер камеры для снижения нагрузки на ЦПУ в Astra Linux.
3. Разработка модуля ядра для зеркального отображения видеопотока камеры средствами драйвера V4L2 в Astra Linux.
4. Создание простого фильтр-драйвера для наложения текстового водяного знака (например, временной метки) на видеопоток в Astra Linux.
5. Модификация драйвера камеры для создания виртуального устройства, выводящего монохромное (черно-белое) изображение в Astra Linux.

Управление устройствами

6. Разработка драйвера для светодиодной подсветки, управляемой через GPIO, и его интеграция с драйвером камеры в Astra Linux.
7. Добавление в драйвер USB-камеры поддержки управления фокусом через стандартный интерфейс V4L2 в Astra Linux.
8. Создание простого драйвера "кнопки" для инициализации захвата кадра по аппаратному сигналу в Astra Linux.
9. Модификация драйвера камеры для сохранения последнего кадра в буфер ядра по внешнему прерыванию в Astra Linux.
10. Реализация в драйвере камеры возможности циклического захвата кадров с заданной частотой в Astra Linux.

Виртуализация и эмуляция устройств

11. Разработка драйвера виртуальной камеры, транслирующего статическое изображение (например, шахматную доску) в Astra Linux.
12. Создание драйвера виртуального устройства, эмулирующего работу камеры с заданным разрешением и частотой кадров в Astra Linux.
13. Модификация драйвера loopback-камеры для передачи видеопотока между процессами в Astra Linux.
14. Разработка простого драйвера, предоставляющего интерфейс V4L2 для чтения изображений из директории на диске в Astra Linux.
15. Создание драйвера виртуального устройства для вывода тестового видео PATTERN (например, цветных полос) в Astra Linux.

Безопасность

16. Реализация в драйвере камеры простого механизма блокировки устройства при превышении порога неудачных попыток открытия в Astra Linux.
17. Модификация драйвера камеры для ведения базового журнала доступа (логирование операций open/close) в Astra Linux.
18. Создание драйвера-фильтра, запрещающего изменение разрешения камеры непривилегированным пользователям в Astra Linux.
19. Добавление в драйвер камеры проверки цифровой подписи загружаемой микропрограммы (на примере заглушки) в Astra Linux.
20. Разработка драйвера виртуальной камеры, транслирующего "заглушку" при невозможности доступа к физическому устройству в Astra Linux.

Вид аттестации: курсовой проект, зачет.

Автор А.С.Жук, доцент КВТ