

ОСНОВЫ АР- ПРИЛОЖЕНИЙ



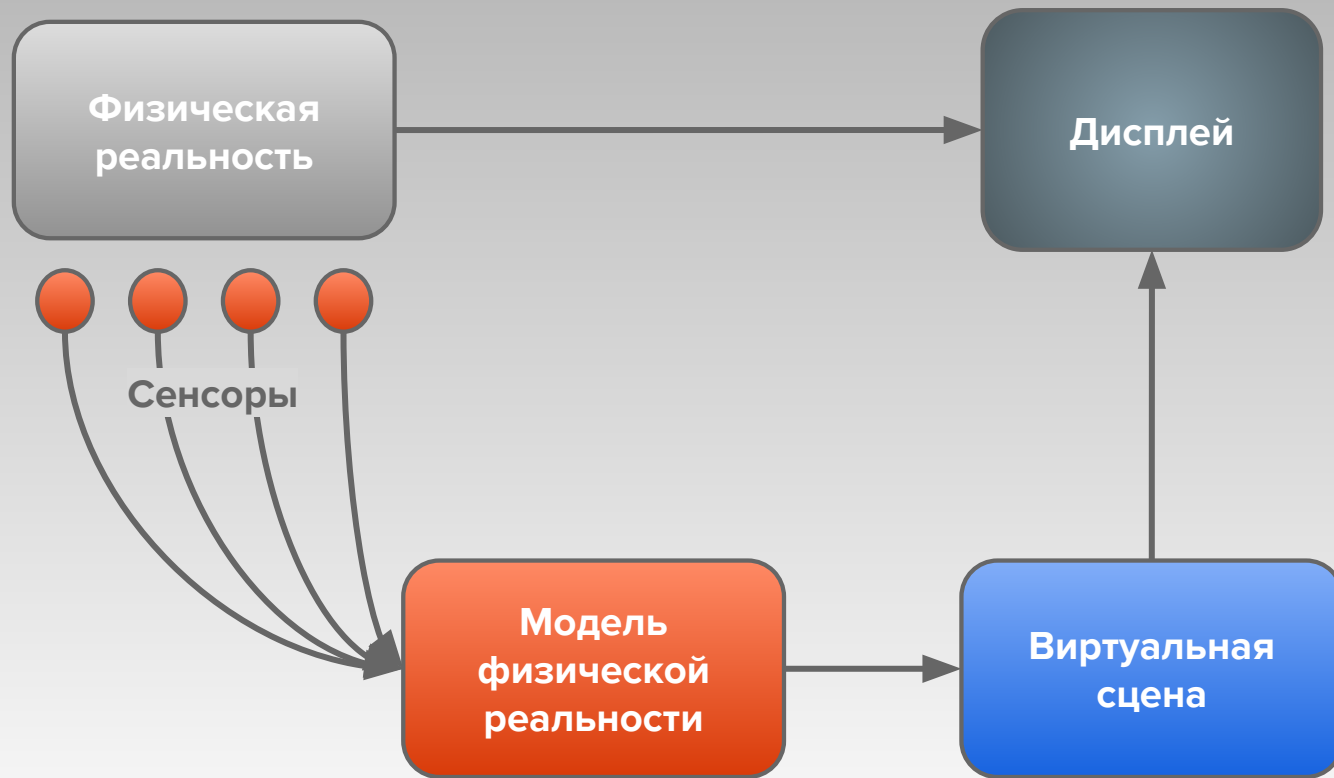
2026

Дополненная реальность

Технология, которая добавляет цифровую информацию к восприятию реального мира, сохраняя его видимым для пользователя.

1992 - Thomas Caudell
использует термин “Augmented Reality”





Архитектура XR-системы

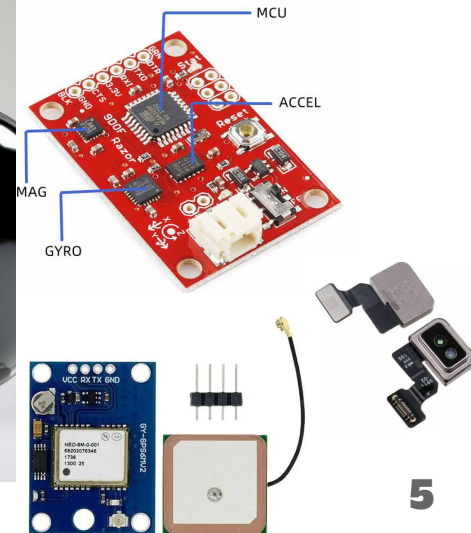
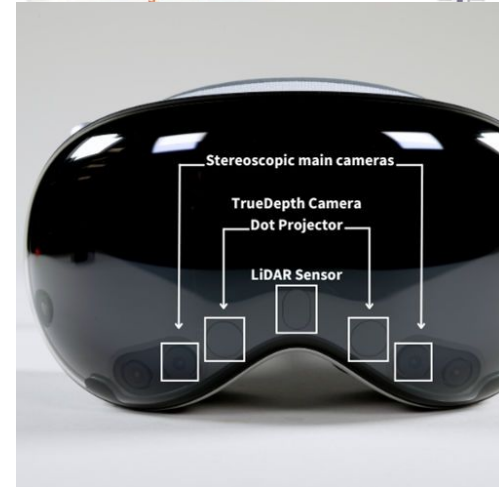
Дисплеи

- Оптические-прозрачные
- Видео-прозрачные
- Проекционные
- Карманные/ручные



Сенсоры

- Камера (-ы)
- Инфракрасные сенсоры
- Лидар системы
- Акселерометр и гироскоп
- Компас и модуль GPS



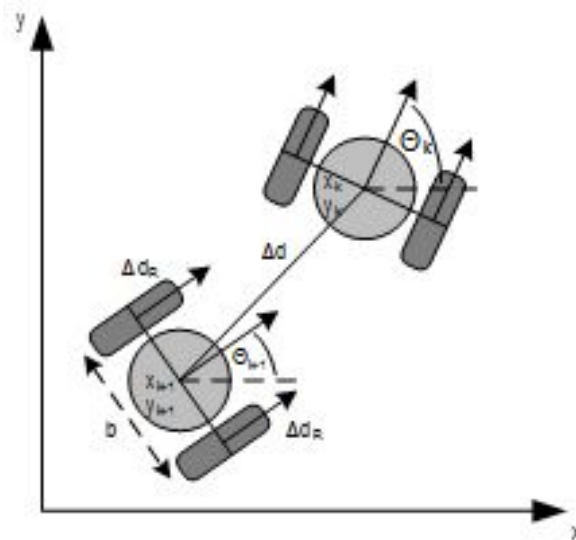
Математические методы в AR

- Одометрия
- Извлечение ключевых точек
- Проективное преобразование
- Извлечение трехмерных параметров сцены
- Трехмерная реконструкция

Одометрия

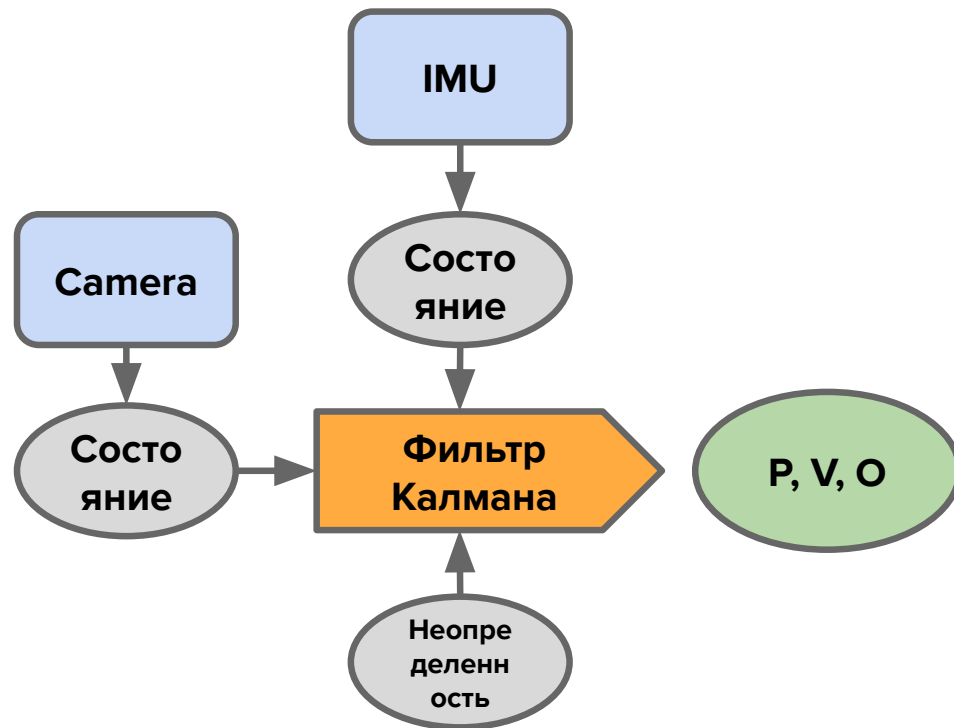
Метод оценки положения и ориентации устройства с помощью анализа данных сенсоров

- Колесная (если у вас есть колеса)
- Инерционная (IMU)
- Визуальная (камера)
- Комбинированная



Visual Inertial Odometry

- Position
- Velocity
- Orientation



Извлечение ключевых точек

1. Методы на основе углов

- Harris
- Shi–Tomasi
- FAST

2. Методы на основе пространства масштабов

- SIFT
- SURF
- KAZE
- BRISK

3. Комбинированные

- ORB = FAST + Harris + BRIEF
- BRISK
- AKAZE

4. Нейросетевые

- SuperPoint
- LIFT
- TILDE
- DetNet

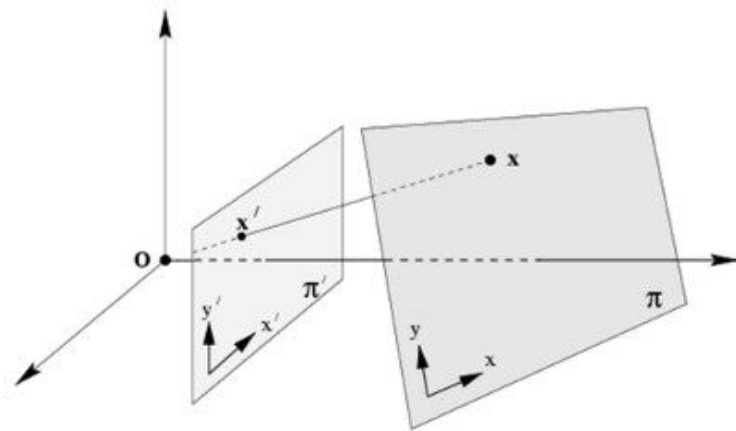
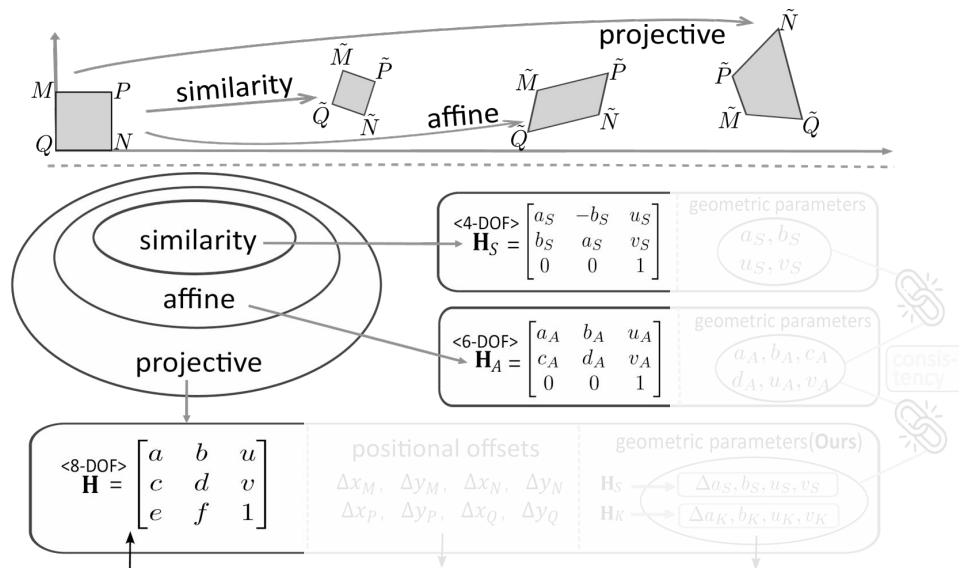
Извлечение ключевых точек \2

Задача	Предпочтительный вариант
Максимальная скорость	FAST
Простой tracking	Harris / FAST
Real-time + хорошее качество	ORB
Масштаб картинок сильно меняется	SIFT / KAZE
Сложные изменения освещения/точки зрения	SuperPoint
Современная система сопоставления изображений	SuperPoint + learned matching

<https://arxiv.org/abs/1908.09300>

Проективное преобразование (Homography)

Взаимно-однозначное отображение проективного пространства на себя, сохраняющее отношение порядка частично упорядоченного множества всех подпространств (прямых, плоскостей)



Использование проективного преобразования

- Извлечение заданных плоскостей
- Вычисление преобразования положения камеры
- Совмещение перспектив



ОПТИЧЕСКИЙ ПОТОК

Отображение видимого движения объектов, поверхностей и границ в сцене, вызванного относительным движением наблюдателя и сцены

$$\frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} = 0$$

Основное уравнение оптического потока содержит две неизвестных и не может быть однозначно разрешено

Метод Лукаса-Канаде

Метод основан на предположении, что в локальной окрестности каждого пикселя значение оптического потока одинаково,

Таким образом можно записать основное уравнение оптического потока для всех пикселей окрестности и решить полученную систему уравнений методом наименьших квадратов

$$\begin{cases} I_x(q_1)V_x + I_y(q_1)V_y = -I_t(q_1) \\ I_x(q_2)V_x + I_y(q_2)V_y = -I_t(q_2) \\ \dots \\ I_x(q_n)V_x + I_y(q_n)V_y = -I_t(q_n) \end{cases}$$

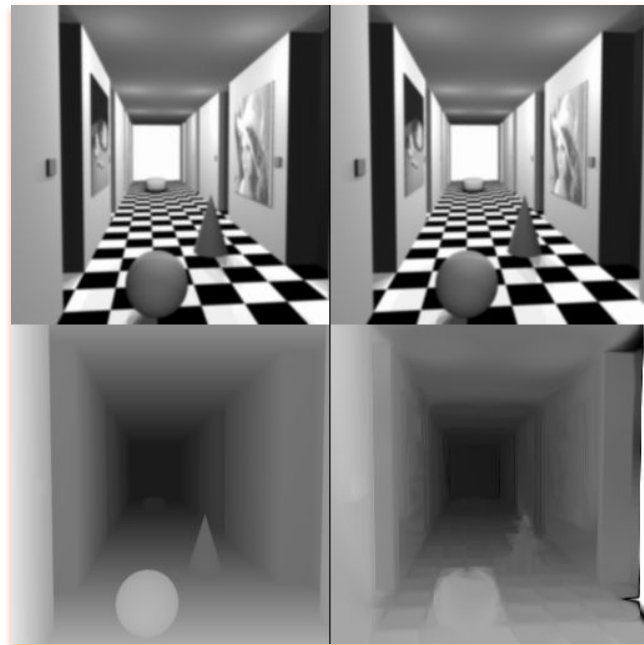
Применение оптического потока

- Сжатие видео
- Стабилизация видео
- Отслеживание объектов
- ***Визуальная одометрия***



Извлечение трехмерных параметров сцены

- Сенсор глубины (IR, Lidar)
- Диспаритет (2 камеры)
- Динамический диспаритет (1 камера)

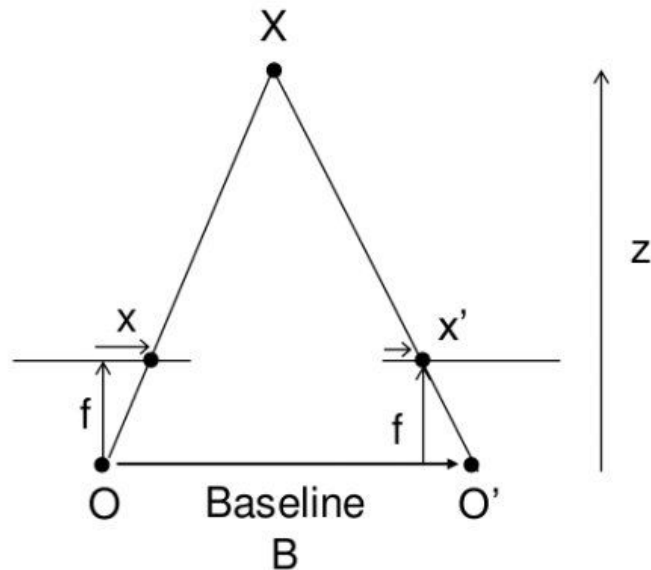


Карта несоответствий (Disparity Map)

При известных параметрах сенсоров:

- угол обзора (fov),
- фокусное расстояние (f),
- расстояние между фокусами (B)

— можно рассчитать приблизительное расстояние до объектов на изображении



Трёхмерная реконструкция

Карта глубины + особые точки = облако точек (x, y, z)

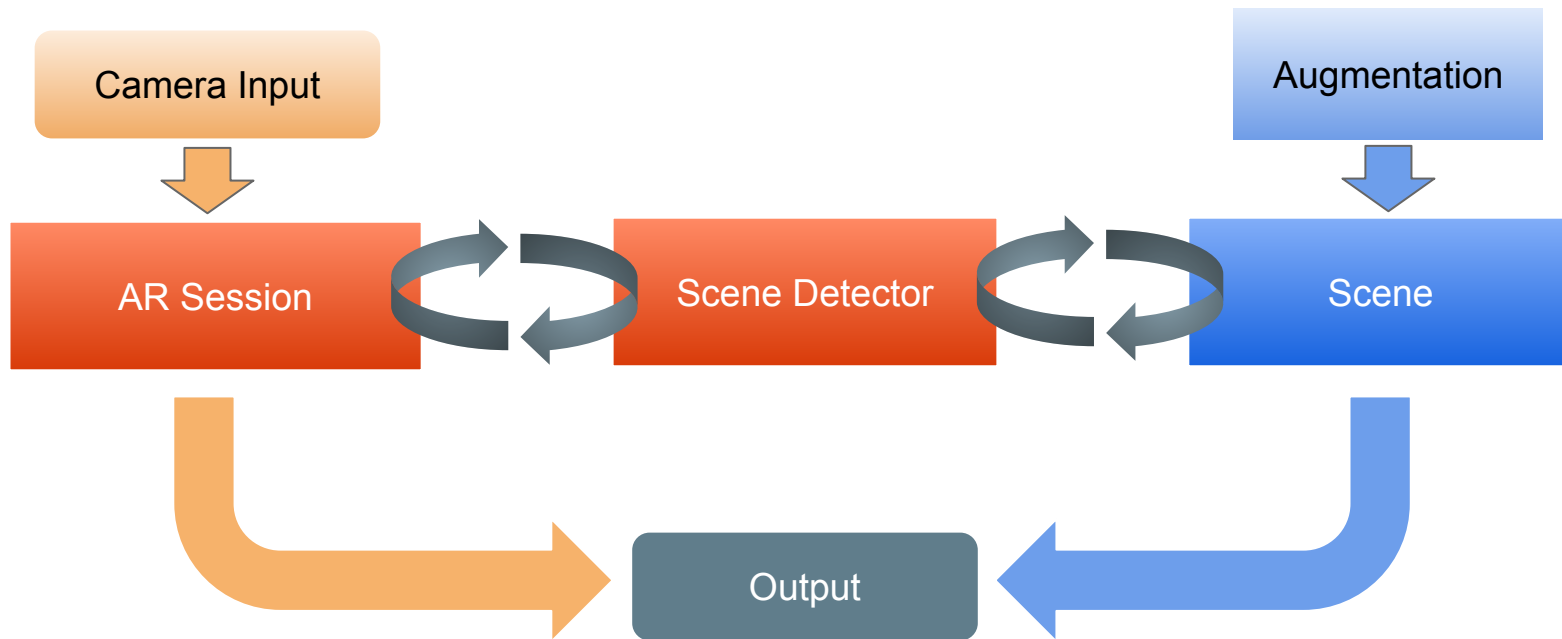
Оценка Sparse Point Cloud для получения геометрии объектов

Метод	Sparse-cloud behavior
Delaunay / 3D Delaunay tetrahedralization	Can produce unwanted tetrahedral connections
Moving Least Squares (MLS) + meshing	Smooths and estimates a surface before triangulation
RBF / implicit surface reconstruction	Can produce continuous surfaces across gaps

Оценка освещения

Method	Lighting Representation	Advantages	Limitations
Parametric Lighting	Point, directional, area lights	Simple, fast, interpretable	Limited for complex lighting
Environment Map Estimation	HDR 2D environment map	Represents complex illumination	Difficult from a single image
Spherical Harmonics (SH)	SH coefficients	Compact and efficient	Poor at sharp light sources
Neural Lighting Estimation	Learned environment/light representation	Handles complex real-world scenes	Requires training data/model

Как это работает



Инструменты

→ AR Core

◆ <https://arvr.google.com/arcore>

→ AR Kit

◆ <https://developer.apple.com/augmented-reality>

→ AR Engine

◆ <https://developer.huawei.com/consumer/en/doc/graphics-Guides/introduction-0000001088644949>

→ Meta XR SDK

◆ <https://developers.meta.com/horizon/develop/>

Сторонние библиотеки

EasyAR

<https://www.easyar.com/>

Niantic

<https://www.nianticspatial.com/docs/>

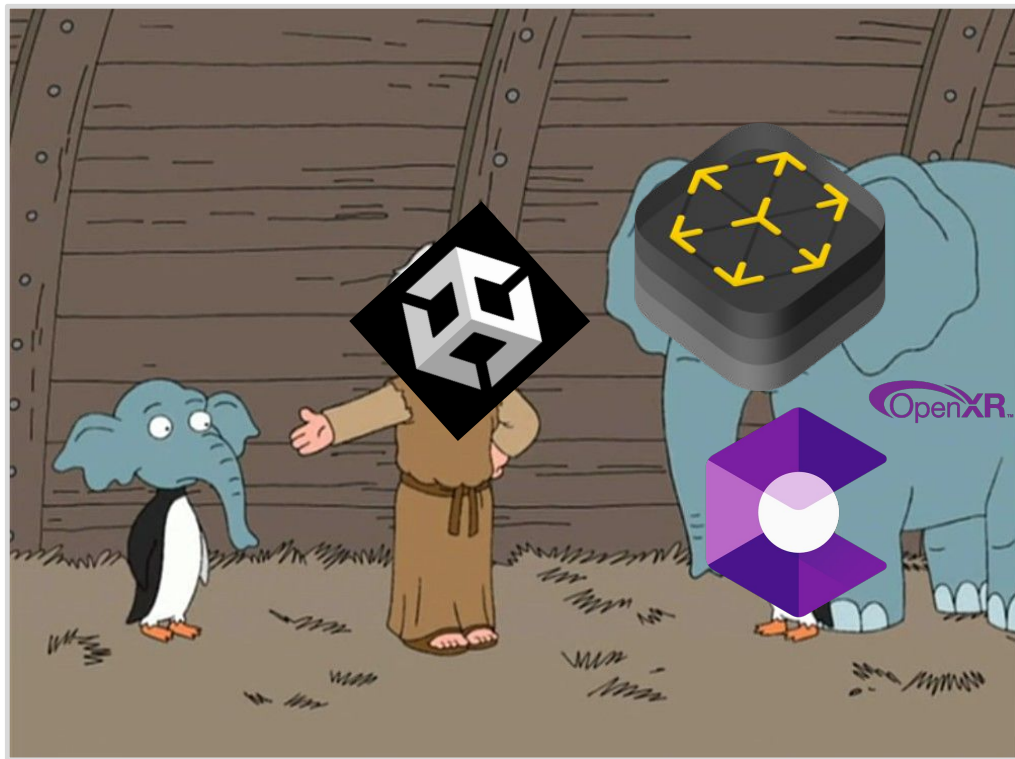
Vuforia

<https://www.ptc.com/en/products/vuforia/>

Web AR

- AFrame
 - ◆ <https://aframe.io>
- ARJS
 - ◆ <https://ar-js-org.github.io/AR.js-Docs/>
- MindAR
 - ◆ <https://github.com/hiukim/mind-ar-js>
- BabylonJS
 - ◆ <https://doc.babylonjs.com/features/featuresDeepDive/webXR/webXRARFeatures/>

Unity AR Foundation



<https://docs.unity3d.com/Manual/AROverview.html>

ОСНОВНЫЕ ВЫЗОВЫ AR

- Распознавание контекстов
- Совместная работа нескольких устройств
- Взаимодействие виртуальных и реальных объектов

