

Настройка OpenCV на Mac в Clion

Установка HomeBrew

Вставьте эту строку в терминал и дождитесь установки HomeBrew:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Что делает HomeBrew? Он устанавливает нужные вам пакеты, не предоставляемые Apple

Установка OpenCV

Введите в терминале: `brew install opencv`

Скачивание OpenCV-contrib

Перейдите по ссылке и скачайте

библиотеку opencv-contrib с GitHub: https://github.com/opencv/opencv_contrib

Запуск программы detect_markers

Создайте проект C++ в Clion

(назовите его detect_markers)

Переименуйте скачанную библиотеку в opencv-contrib и перетащите её в папку проекта detect_markers в Finder

В CmakeLists пропишите следующий код:

```
cmake_minimum_required(VERSION 3.21)
project(opencv_example)
set(CMAKE_CXX_STANDARD 14)
find_package(OpenCV REQUIRED)
include_directories(${OpenCV_INCLUDE_DIRS} opencv_contrib/modules/aruco/samples)
add_executable(opencv_example main.cpp)
target_link_libraries(opencv_example ${OpenCV_LIBS})
```

Затем идём в папку

opencv-contrib/modules/aruco/samples и открываем файл detect_markers.cpp

Копируем код из detect_markers.cpp в main.cpp

Компилируем и должно появиться в консоли следующее:

```
Basic marker detection
Usage: opencv_example [params]

-c
    Camera intrinsic parameters. Needed for camera pose
--cd
    Input file with custom dictionary
--ci (value:0)
    Camera id if input doesnt come from video (-v)
-d
    dictionary: DICT_4X4_50=0, DICT_4X4_100=1, DICT_4X4_250=2, DICT_4X4_1000=3, DICT_5X5_50=4, DICT_5X5_100=5, DICT_5X5_250=6, DICT_5X5_1000=7, DICT_6X6_50=8,
    DICT_6X6_100=9, DICT_6X6_250=10, DICT_6X6_1000=11, DICT_7X7_50=12, DICT_7X7_100=13, DICT_7X7_250=14, DICT_7X7_1000=15, DICT_ARUCO_ORIGINAL = 16,
    DICT_APRILTAG_16h5=17, DICT_APRILTAG_25h9=18, DICT_APRILTAG_36h10=19, DICT_APRILTAG_36h11=20
--dp
    File of marker detector parameters
-l (value:0.1)
    Marker side length (in meters). Needed for correct scale in camera pose
-r
    show rejected candidates too
--refine
    Corner refinement: CORNER_REFINE_NONE=0, CORNER_REFINE_SUBPIX=1, CORNER_REFINE_CONTOUR=2, CORNER_REFINE_APRILTAG=3
-v
    Input from video or image file, if omitted, input comes from camera

Process finished with exit code 0
```

Заходим в **Terminal** в Clion (расположен внизу) и пишем: **./detect_markers -d**

```
vsevolod@MacBook-Air-Vsevolod cmake-build-debug % ./detect_markers -d
```

Version Control Run TODO Problems Terminal CMake Python Packages Messages

**Должна запускаться камера
(если был запрошен доступ
к камере, то снова запускаем
программу в терминале)**

**Для демонстрации работы
покажите в камеру телефон
с маркером с сайта <https://chev.me/arucogen/>**