**Документация**

***Выполнил: Атай Султангазиев***

import UIKit

import AVKit

import Vision

Импортируем стандартную библиотеку UIKit, AVKit, Vision

let captureSession = AVCaptureSession()

captureSession.sessionPreset = .photo

Говорим, что источник информации – задняя камера.

captureSession.startRunning()

Запускаем сессию

let dataOutput = AVCaptureVideoDataOutput()

dataOutput.setSampleBufferDelegate(self, queue: DispatchQueue(label: "videoQueue"))

captureSession.addOutput(dataOutput)

Выводим уже обработанную информацию в консоль

func captureOutput(\_ output: AVCaptureOutput, didOutput sampleBuffer: CMSampleBuffer, from connection: AVCaptureConnection) {

guard let pixelBuffer: CVPixelBuffer = CMSampleBufferGetImageBuffer(sampleBuffer) else { return }

// !!!Important

// make sure to go download the models at https://developer.apple.com/machine-learning/ scroll to the bottom

guard let model = try? VNCoreMLModel(for: Resnet50().model) else { return }

let request = VNCoreMLRequest(model: model) { (finishedReq, err) in

//perhaps check the err

// print(finishedReq.results)

guard let results = finishedReq.results as? [VNClassificationObservation] else { return }

guard let firstObservation = results.first else { return }

print(firstObservation.identifier, firstObservation.confidence)

DispatchQueue.main.async {

self.identifierLabel.text = "\(firstObservation.identifier) \(firstObservation.confidence \* 100)"

}

}

try? VNImageRequestHandler(cvPixelBuffer: pixelBuffer, options: [:]).perform([request])

}

Функция captureOutput вызывается 2 раза в секунду и обрабатывает информацию с камеры.