**Ver. 8.0.0**

SelvasAI SelvyOCRforMobileScan iOS 개발자 가이드

SelvasAI SelvyOCRforIDCard iOS 개발자 가이드

**2018.09.17**

**㈜ SelvasAI**

목차

[**1. 개요 3**](#_Toc524951696)

[**2. 실행 환경 3**](#_Toc524951697)

[**3. 기능 소개 3**](#_Toc524951698)

[**4. API Reference 4**](#_Toc524951699)

[**5. SDK 구성 4**](#_Toc524951700)

[**6. 예제에 대한 설명을 통한 개발 가이드 4**](#_Toc524951701)

1. 개요

본 문서는 SelvyOCRforMobileScan SDK에 대한 기술적인 정보를 제공하고 SDK를 사용하여 어플리케이션에 문서 영역 검출 및 이미지 처리 기능을 빠르게 구현할 수 있는 방법을 설명한다.

1. 실행 환경

SDK는 다음과 같은 실행 조건에서 동작하도록 개발되었다.

* **운영 체제 : iOS 7.0 이상의 ARM CPU 기반 장치**
* **인식 대상 : A4 문서**
* **런타임 메모리(RAM) : 25MB ~ 100MB(iPhone 6S Plus 기준)**
* **예제 개발 환경 : Xcode 9.4.1, OBJC**

1. 기능 소개

SDK는 다음과 같은 기능을 어플리케이션에 제공한다.

* **촬영 및 문서 영역 검출**

**어플리케이션이 문서 촬영 및 영역 검출을 요청하면 SDK의 촬영 모듈은 단말기의 카메라를 구동하며, 이후 자동 혹은 수동 촬영 방식으로 촬영된 이미지 내에서 문서 영역을 검출한다.**

* **문서 보정 처리**

**SDK의 영상 처리 모듈은 촬영된 이미지의 원근 왜곡, 기울어짐 등을 파악하며, 이를 보정한 이미지를 생성한다.**

* **이미지 처리**

**SDK의 이미지 처리 모듈은 촬영된 이미지의 대한 Contrast/Brightness, Soften, Sharpen, Binarization, TIFF 포맷 저장 등의 처리 기능을 제공한다.**

1. API Reference

SDK의 주요 API에 대한 설명은 별도로 첨부된 API Reference 문서를 참고한다.

1. SDK 구성

SDK는 본 개발자 가이드, 별첨된 API Reference, 그리고 아래와 같은 라이브러리와 예제 어플리케이션으로 구성된다.

* **Sample**
* **sample/SelvyOCRforMobileScanDemo**

**문서 영역 검출 샘플 프로젝트(Xcode 9.4.1, OBJC)**

* **sample/SelvyOCRforMobileScanDemo/Assets.xcassets**

문서 영역 검출 샘플 프로젝트에서 사용되는 이미지들 모음

* **sample/SelvyOCRforMobileScanDemo/ViewControllers/Document**

문서 촬영 및 영역 검출의 모든 과정을 처리하는 예제 ViewController

* **sample/SelvyOCRforMobileScanDemo/ViewControllers/ImageCorrection**

이미지 처리 및 TIFF 포맷 저장 관련 예제 ViewController

* **Library**
  + - **sample/SelvyOCRforMobileScanDemo/libs/SelvyMobileScan**

**문서 영역 검출 라이브러리 및 헤더(armv7/arm64)**

* + - **sample/SelvyOCRforMobileScanDemo/libs/SelvyImageProcessing**

문서 영역 검출 및 이미지 처리에 사용되는 라이브러리 및 헤더(armv7/arm64)

1. 예제에 대한 설명을 통한 개발 가이드

SDK가 제공하는 촬영 및 인식 기능을 어플리케이션에 적용하는 방법은 다음과 같은 절차를 따른다.***(예제 프로젝트 구성 참고***)

1. 라이브러리(엔진)에서 필요로 하는 프레임워크 추가
   1. AVFoundation.framework
      1. 프로젝트 속성의 “BuildPhases->Link Binary With Libraries”에서 추가
   2. CoreMedia.framework
      1. 프로젝트 속성의 “BuildPhases->Link Binary With Libraries”에서 추가
   3. AssetsLibrary.framework
      1. 프로젝트 속성의 “BuildPhases->Link Binary With Libraries”에서 추가
2. Privacy 권한 추가(iOS 10 이상 feature)
   1. Privacy - Camera Usage Description
      1. Project navigator의 info.plist에 “Privacy - Camera Usage Description” 추가
3. 라이브러리(엔진) 추가
   1. SelvyOCR 추가
      1. SelvyMobileScan 라이브러리(.a, .h)를 사용하고자 하는 프로젝트에 추가
      2. SelvyImageProcessing 라이브러리(.a, .h)를 사용하고자 하는 프로젝트에 추가
4. 이미지 리소스 추가(선택사항)
   1. 예제에서 사용되는 이미지 추가
      1. sample/SelvyOCRforMobileScanDemo/Assets.xcassets 폴더 내의 “btn\_camera\_shutter.imageset” 폴더를 사용하고자 하는 프로젝트 내의 Assets.xcassets에 추가
5. ViewController 추가(선택사항, 개발자 커스터마이징)
   1. Document
      1. sample/SelvyOCRforMobileScanDemo/ViewControllers/Document 의 모든 파일을 사용하고자 하는 프로젝트에 추가
   2. ImageCorrection
      1. sample/SelvyOCRforMobileScanDemo/ViewControllers/ImageCorrection 의 모든 파일을 사용하고자 하는 프로젝트에 추가
6. 문서 영역 검출 및 결과 처리
   1. 촬영 및 영역 검출 요청
      1. sample/SelvyOCRforMobileScanDemo/ViewController.mm 참고
      2. 자동/수동 선택 후 “촬영” 버튼 누를 시, “tapGoCapture” instance method를 통해 DocumentVC 호출
      3. Delegation(영역 검출 결과)을 받기 위해 SelvyDocumentsDetectorDelegate”를 재정의 해야함
      4. DocumentVC와 예제 ViewController는 rootViewController에 present/dismiss 되는 형태
   2. 인식 결과 처리
      1. sample/SelvyOCRforMobileScanDemo/ViewController.mm 참고
      2. 촬영화면에서 촬영 후 영역 검출이 완료되면 미리 정의 해둔 delegate를 통해 callback을 받는 방식