



Clova Face Kit: 10분 안에 누구나 적용하는 얼굴인식

강정현 정상엽 김범수 유준희
NAVER Clova

CONTENTS

1. Introduction

2. Hands-On

2.1. Settings & Options

2.2. iOS

2.3. Android

2.4. Python

2.5. JavaScript

3. Closing

1. Introduction

1.1. Clova Face Kit?

단말(edge)에서 구동되는 얼굴 인식 기술

Total Solution
for face & human analysis

No need for **expensive GPU**
server

Support **cross-platform**

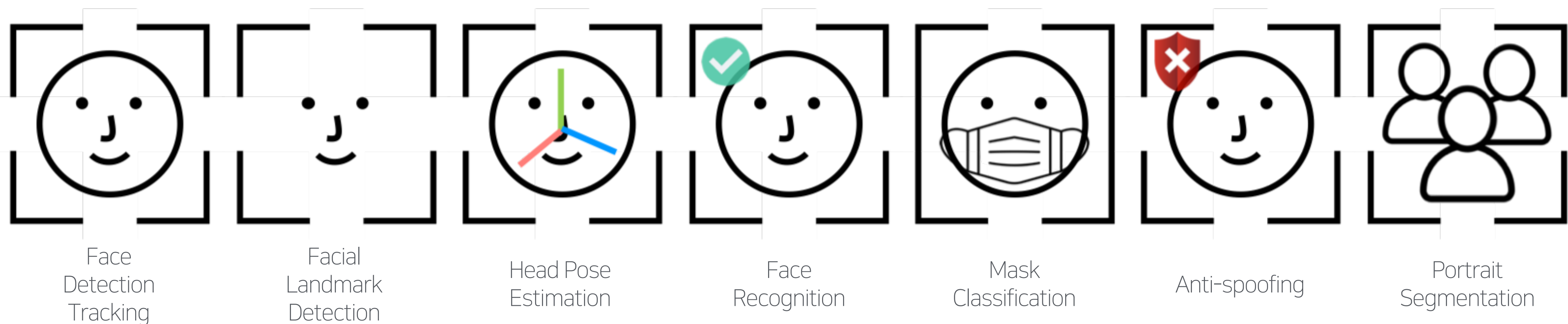
Easy & Simple
API

1.2. 지원 기능

다양한 플랫폼 지원: Android, iOS, Linux, macOS, Windows, WebAssembly

다양한 인터페이스 지원: C++, JavaScript, Kotlin, Objective-C, Python

다양한 기능 지원:



1.3. 사용자

- 강남언니
- Clova FaceSign
- LINE eKYC
- NAVER 포토클라우드

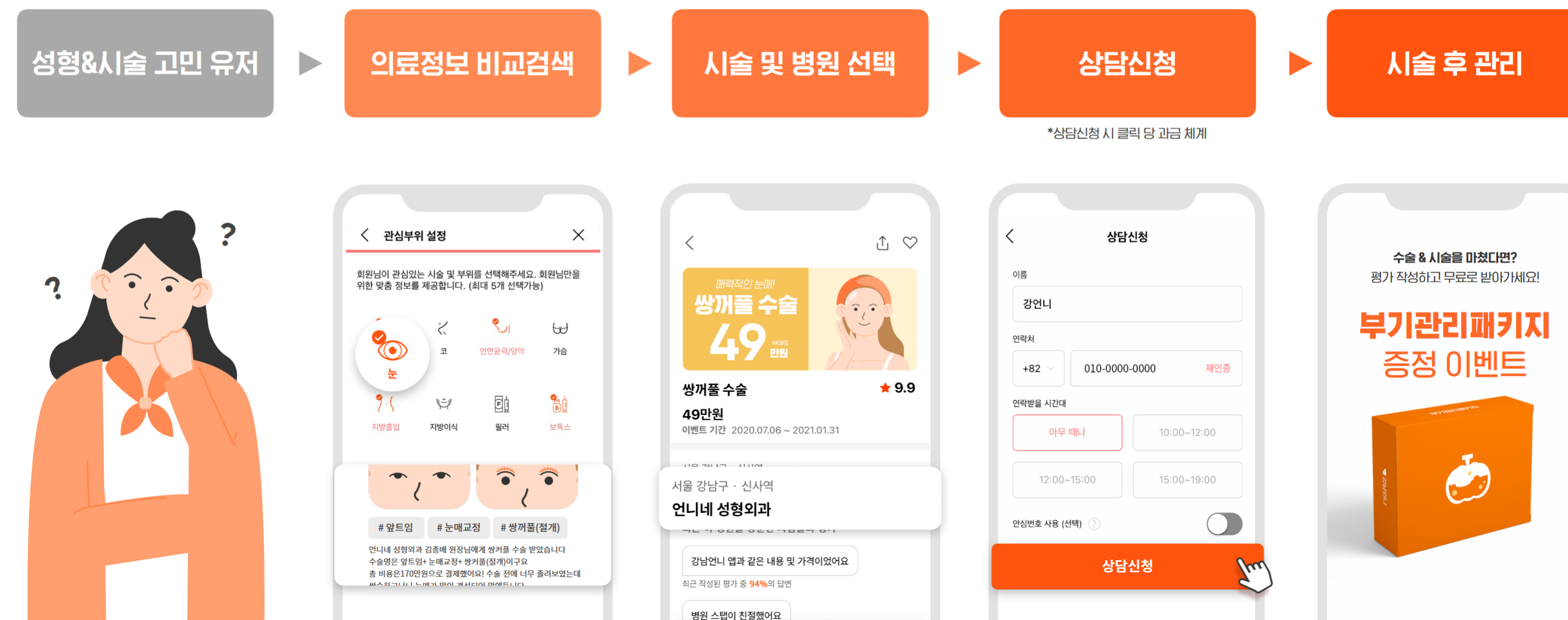
1.3. 강남언니

- 공급자 중심의 의료서비스를
- IT 기반 미용의료 정보 플랫폼을 통해
- 의료소비 주체가 중심 되는 시장으로

강남언니

언니를 만나고 **병원선택** 고민은 끝났다

강남언니는 시술 및 병원 선택부터 상담신청, 시술 후 관리까지 **유저의 고민 해결 여정**을 돕는 플랫폼입니다.



1.3. 강남언니 시술 후기

- 다른 사용자의 시술 후기는 시술 상품 선택에 큰 도움이 됨
- 사진이 포함된 후기는 신뢰도가 더욱 높아짐
- 사용자는 후기에 사진을 첨부할 때 일부 영역을 감추고 싶은 요구를 가짐
- 기존에는 각 사용자가 직접 각종 도구를 사용해 사진을 편집

1.3. Clova Face Kit 사용 후기 사진 보정



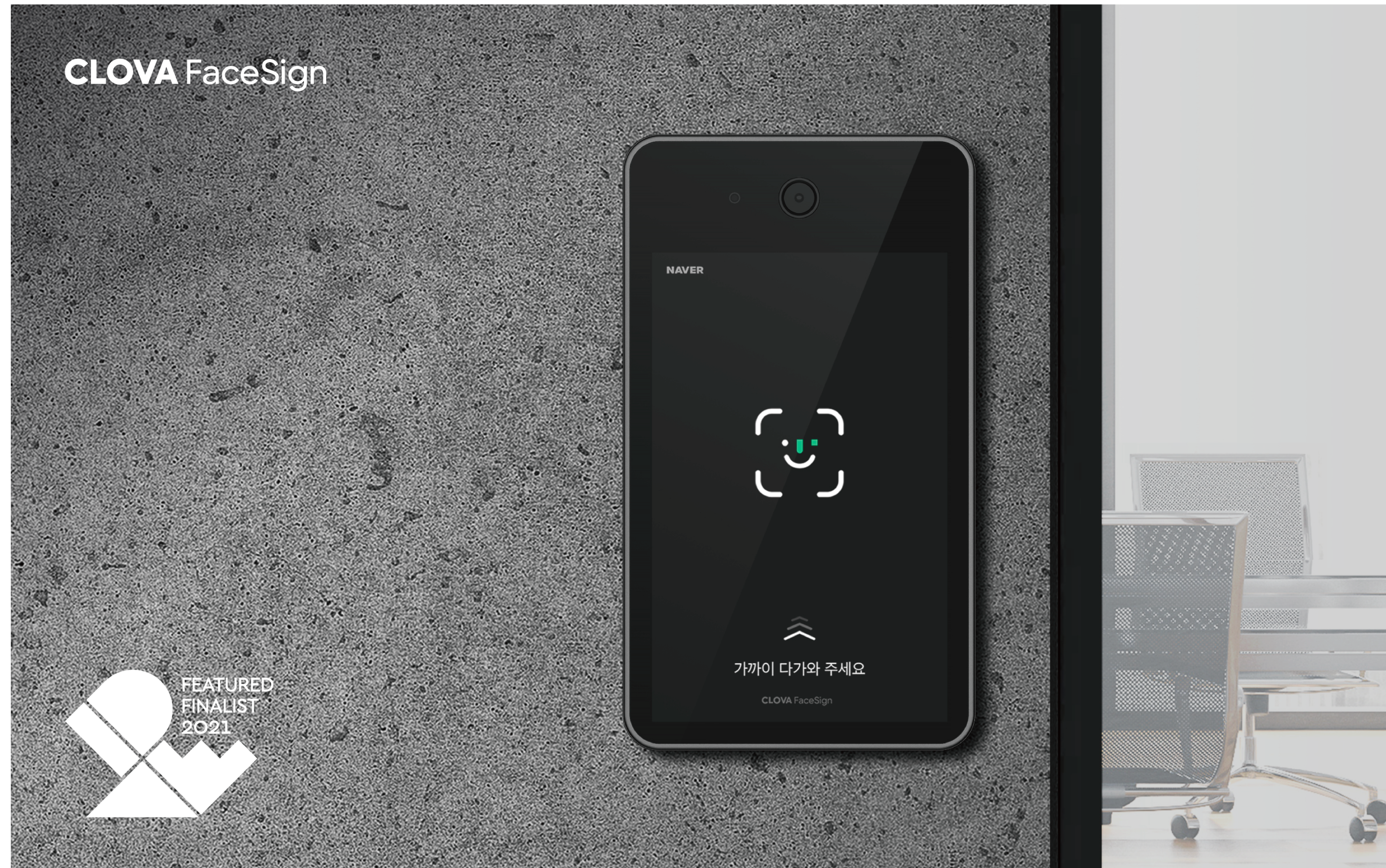
Clova
Face Kit



1.3. 강남언니의 Clova Face Kit 효과

- 사용자는 민감한 얼굴 사진 중 후기에 해당하는 부분만 강남언니 서비스에 제공
- 강남언니 시스템 자원 대신 사용자의 단말기 자원 사용
- 일관된 사진 보정 효과
- 언제나 인적 자원이 부족한 스타트업이 준비된 도구를 사용해 매우 짧은 시간에 서비스 가치 창출

1.3. Clova FaceSign



1.3. NAVER 포토 클라우드

MYBOX

MYBOX 바로가기 이용권 공식 블로그

네이버 MYBOX와 함께 소중한 사진, 자료를 한 곳에서

국내 최대 무료 용량 30GB로 시작하세요.
사진과 문서 자료를 한 곳에 안전하게 보관하고 작업할 수 있습니다.
강력한 사진 정리와 감상, 문서 조회와 편집, 공유 기능을 제공합니다.

[모바일 앱 다운로드](#) [데스크톱 앱 다운로드](#)

[새 소식](#) [영상 보기](#) [기능 소개](#)

1.3. LINE eKYC

LINE eKYC

LINE eKYCとは

導入事例

ユースケース

特長

導入の流れ

LINE eKYCのお問い合わせ

LINEが開発した高精度のeKYCソリューション
安心・安全×スムーズなオンライン本人確認

LINE eKYC

資料請求・お問い合わせはこちら



2.Hands-On

2. Hands-On

1. Settings & Options
2. iOS
3. Android
4. Python
5. JavaScript

2.1. Settings & Options

2.1. Settings & Options

Notice

- Clova Face Kit은 구 명칭인 ClovaSee로 부터 명칭을 변경하는 중입니다.
- Hands-On의 예제 곳곳에 ClovaSee의 흔적이 남아 있으나, 추후 변경 예정입니다.

2.1. Settings & Options

Settings: Clova Face Kit instance에 영향을 미치는 설정 값

Options: Clova Face Kit의 Frame 처리 1회에 대한 설정

- Platform 혹은 언어별로 명명 형식이 상이함. (camel case, snake case 등)
- Settings와 Options들의 값과 효과는 모두 동일함.
 - C/C++을 기반으로 하기 때문

C/C++

iOS

Android

Python

JavaScript

2.1. Settings

모든 Settings는 SettingsBuilder로 부터!

예시)

```
Settings s = SettingsBuilder().setIntermittentInformationRatio(v)
                                .setNumberOfThreads(n)
                                .build()
```

2.1. Settings

`SetIntermittentInformationRatio(uint8_t value);`

- Clova Face Kit의 정보(특징점, 각도 등)을 몇 Frame마다 생성할지 설정
- 10을 설정하는 경우, 0~9 Frame에 대해서는 정보를 생성하지 않음.
- 얼굴을 찾는 것(bounding box)은 예외.
- Default: 1

`SetNumberOfThreads(int value)`

- Thread를 최대 몇개까지 사용할지 설정
- Default: 4

2.1. Settings

`SetPerformanceMode(const Settings::PerformanceMode& value)`

- 얼굴의 특징점을 찾는 방식을 설정.
- kAccurate106, kAccurate98, kFast 값을 사용 가능.

2.1. Options

모든 Options는 OptionsBuilder로 부터

```
Options op =  
OptionsBuilder().OptionsSetInformationToObtain(v)  
    .SetSmoothingRect(b)  
    .build()
```

2.1. Options

`SetBoundingBoxThreshold(float value)`

- 얼굴 탐지결과 반환시 confidence가 value 이하의 결과값을 가진 bounding box들을 사용하지 않습니다.

`SetMinimumBoundingBoxSize(float value)`

- Frame을 기준으로 얼굴인식 결과의 최소 크기를 설정합니다.
- Value보다 작은 얼굴은 인식결과에서 제외합니다.

2.1. Options

SetResizeThreshold(`int value`)

- 주어진 Frame의 처리를 위해 Resize할 때 장축의 길이를 설정합니다. 예를 들어 값이 320이고 Frame의 크기가 620, 100이었다면, 내부 Resizing에서 Frame의 크기는 320, 50으로 변경됩니다.
- Default: 320

SetSmoothingRect(`bool value`)

- Frame의 변화에 따라 Bounding box들이 변화할 때 튀는 정도를 부드럽게 바꿉니다.
- Default: false

SetSmoothingContour(`bool value`)

- Frame의 변화에 따라 Lamdmak들이 변화할 때 튀는 정도를 부드럽게 바꿉니다.
- Default: true

2.1. Options

SetInformationToObtain(uint8_t value)

- Bounding box외에 추가로 얻을 정보를 flag로 전달합니다.
- kBoundingBoxes
얼굴이 있는 영역의 사각형 정보를 반환합니다.
- kContours
얼굴의 윤곽선 정보를 추가하여 반환합니다.
- kMasks
마스크 착용 여부를 추가하여 반환합니다.
- kTrackingIDs
얼굴에 추적 ID를 추가하여 반환합니다.
- kSpoofs
얼굴이 진짜 얼굴인지 여부를 추가하여 반환합니다.
- kEulerAngles
얼굴의 각도: Yaw, Roll, Pitch 값을 추가하여 반환합니다.
- kAll
위 정보를 모두 추가하여 반환합니다.

2.2. iOS Hands-On

2.2. iOS Hands-On

naver / clova-face-kit Public Notifications Star 22 Fork 1

Code Issues Pull requests Actions Wiki Security Insights

main clova-face-kit / examples / ios / Go to file

정상엽 and YoungjaeKim modify README.md contents in examples/ios 379e513 8 days ago History

..		
Sample_Face.xcodeproj	update iOS Demo & quick_developer_guide_ios_ko.md	8 days ago
Sample_Face	update iOS Demo & quick_developer_guide_ios_ko.md	8 days ago
.gitignore	Initial commit	last month
README.md	modify README.md contents in examples/ios	8 days ago

README.md

How to build

- see the iOS guide [link](#)

2.2. iOS Hands-On

Latest release

0.2130
175385f

Compare ▾

0.2130

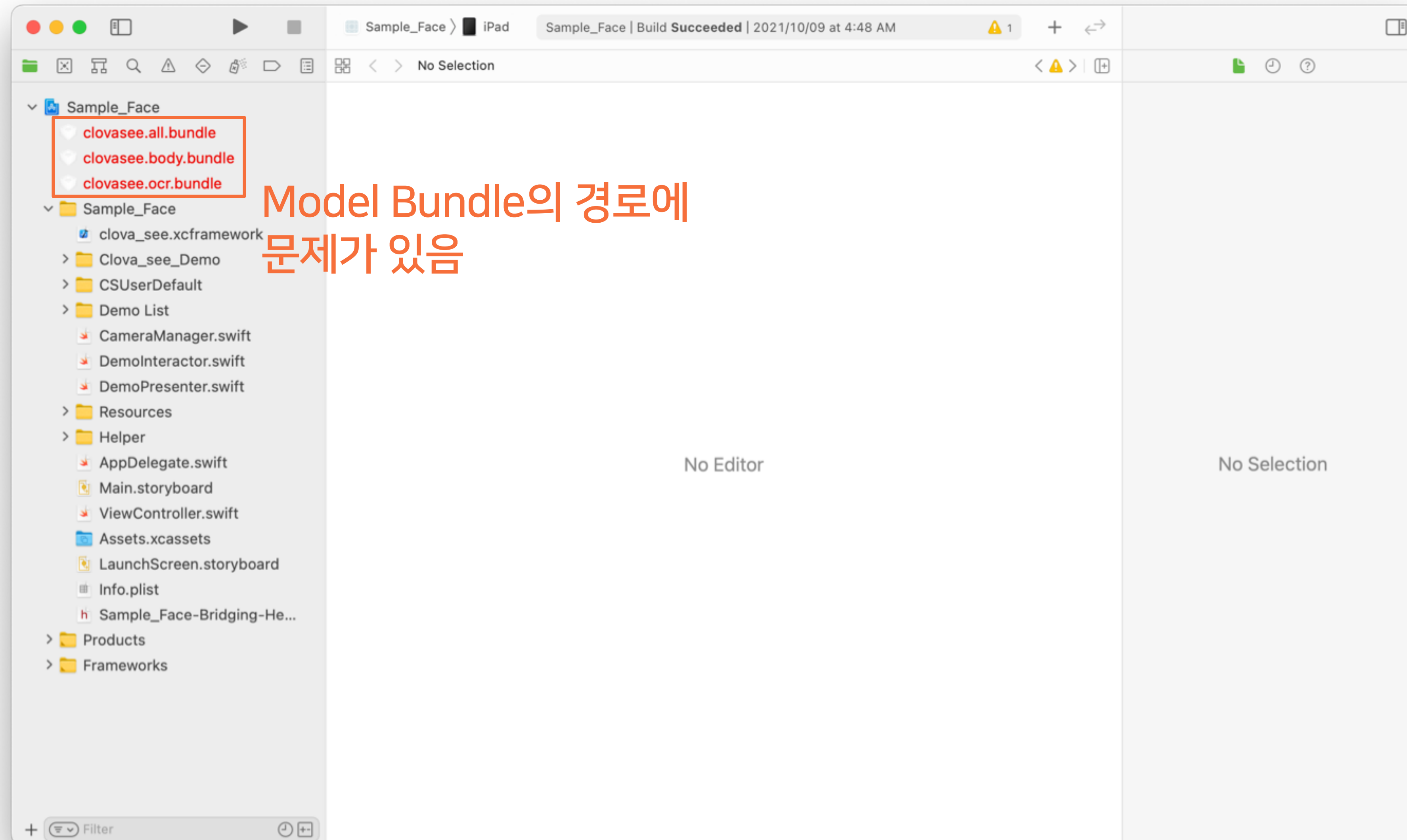
junhee-yoo released this on 17 Sep

Released

Assets 18

clova-see-0.2130.54.aar	30.2 MB
clovasee-0.2130.54-cp38-cp38-win_amd64.whl	20 MB
clovasee-0.2130.60-cp36-cp36m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp37-cp37m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp38-cp38-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-Linux.sh	3.32 MB
clovasee-0.2130.70-cp38-cp38-macosx_10_14_x86_64.whl	20.6 MB
clovasee-0.2130.70-cp39-cp39-macosx_10_15_x86_64.whl	20.6 MB
clovasee-0.2130.70-Darwin.sh	622 KB
clovasee-0.2930.0-webassembly.zip	19.6 MB
clovasee.all.bundle	20.8 MB
clovasee.body.bundle	188 KB
clovasee.face.bundle	22 MB
clovasee.face_without_spoofing_detection.bundle	6.6 MB
clovasee.ocr.bundle	3.3 MB
clova_see.xcframework-0.2130.45.zip	3.84 MB
Source code (zip)	
Source code (tar.gz)	

2.2. iOS Hands-On



Model Bundle의 경로에 문제가 있음

2.2. iOS Hands-On

1. 유효하지 않은 경로를 가리키던 bundle을 모두 지움
 2. clovasee.all.bundle을 project에 추가
 3. Finder에서 clova_see.xcframework를 다운받은 것으로 교체

2.2. iOS Hands-On

The screenshot shows the Xcode IDE interface. On the left, the Project Navigator displays a project named 'Sample_Face' with a sub-project 'clovasee.all.bundle'. The main canvas area contains a large, 3D-rendered white cube. At the bottom center, a white notification bubble with a hammer icon and the text 'Build Succeeded' is visible. On the right, the Properties Inspector shows details for the selected bundle, including its name, type, location, and target membership.

Identity and Type

- Name: clovasee.all.bundle
- Type: Default - Data
- Location: Relative to Group
- Full Path: /Users/user/clova-face-kit/examples/ios/clovasee.all.bundle

On Demand Resource Tags

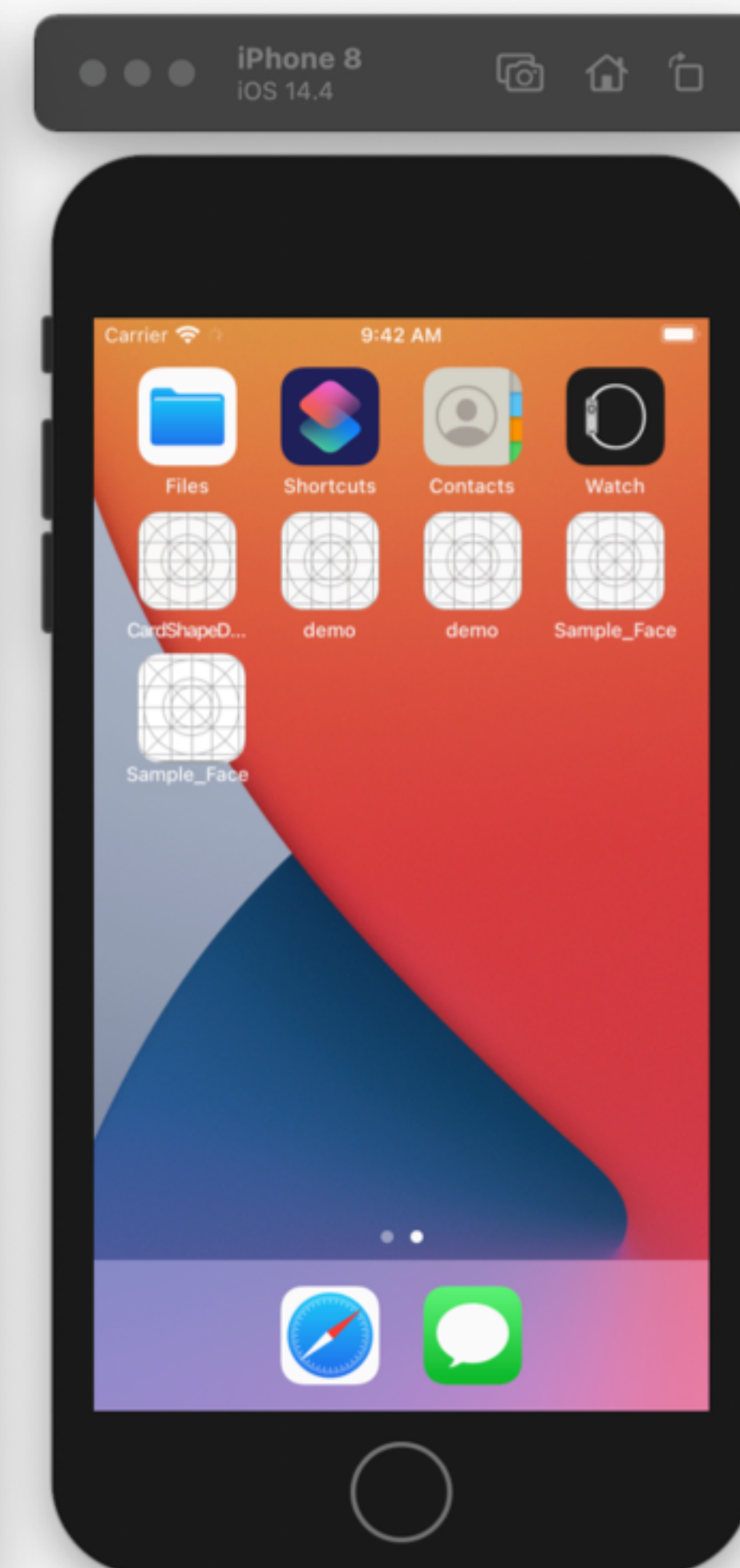
Tags

Localization

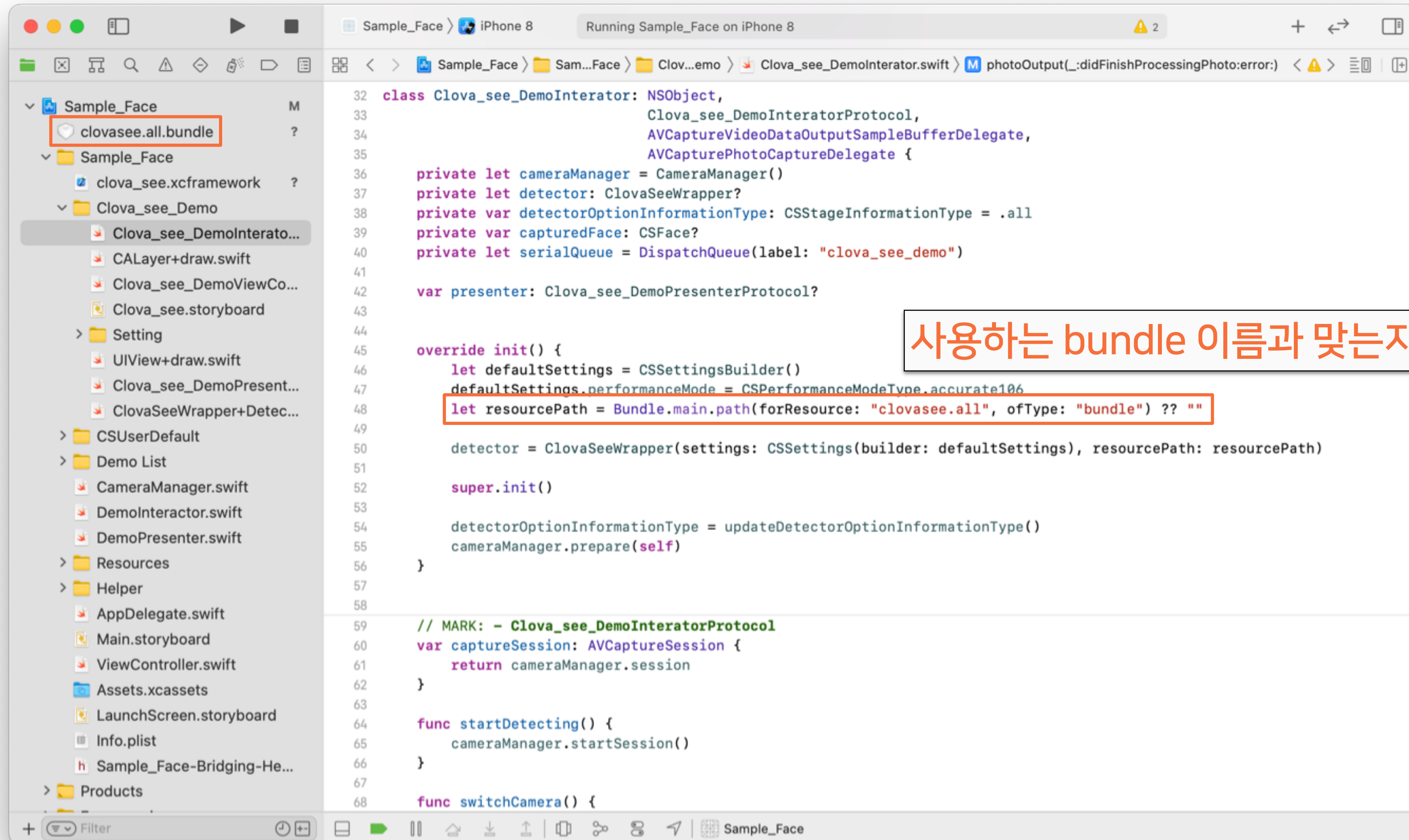
Localize...

Target Membership

- Sample_Face

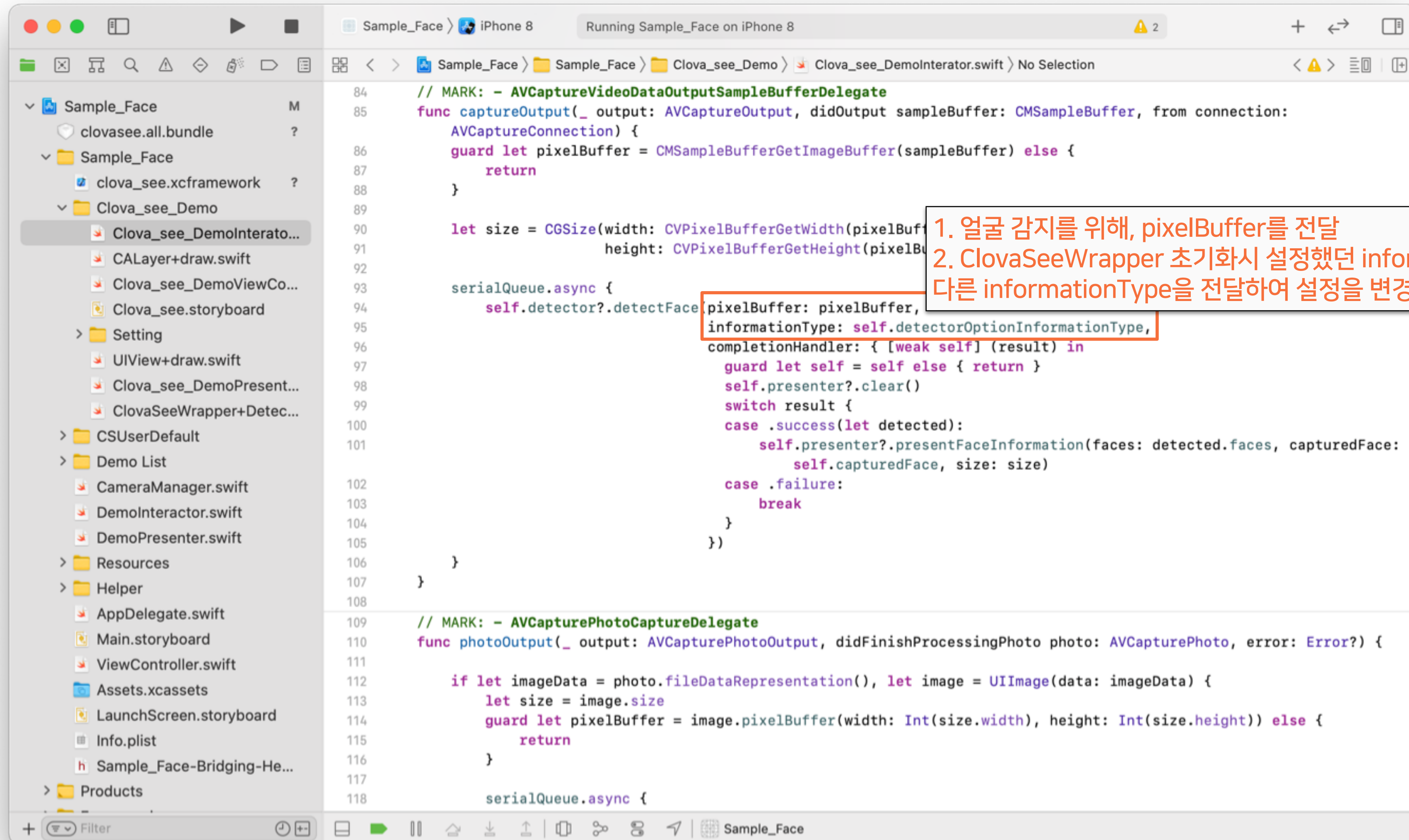


2.2. iOS Hands-On

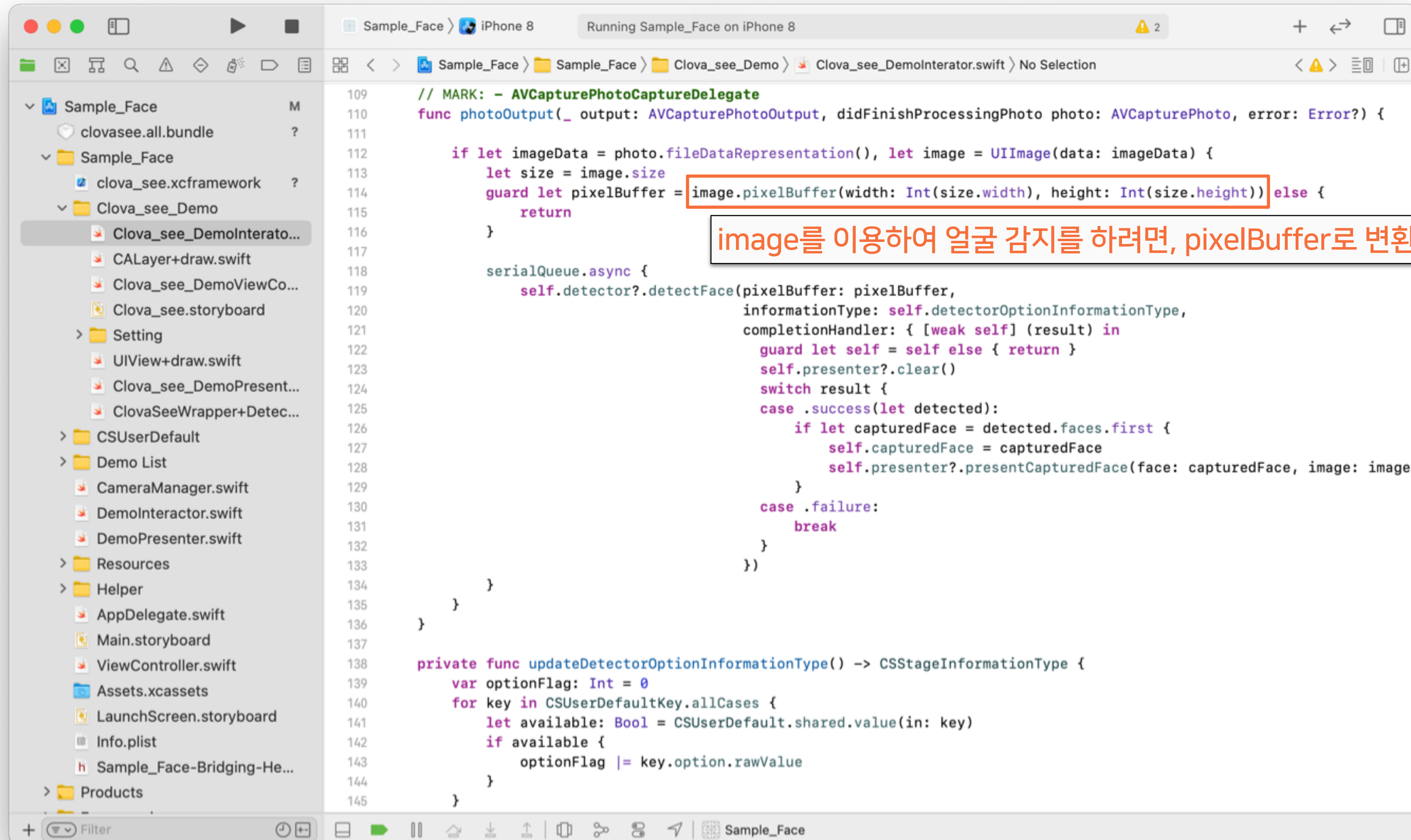


사용하는 bundle 이름과 맞는지 확인

2.2. iOS Hands-On



2.2. iOS Hands-On



2.2. iOS Hands-On

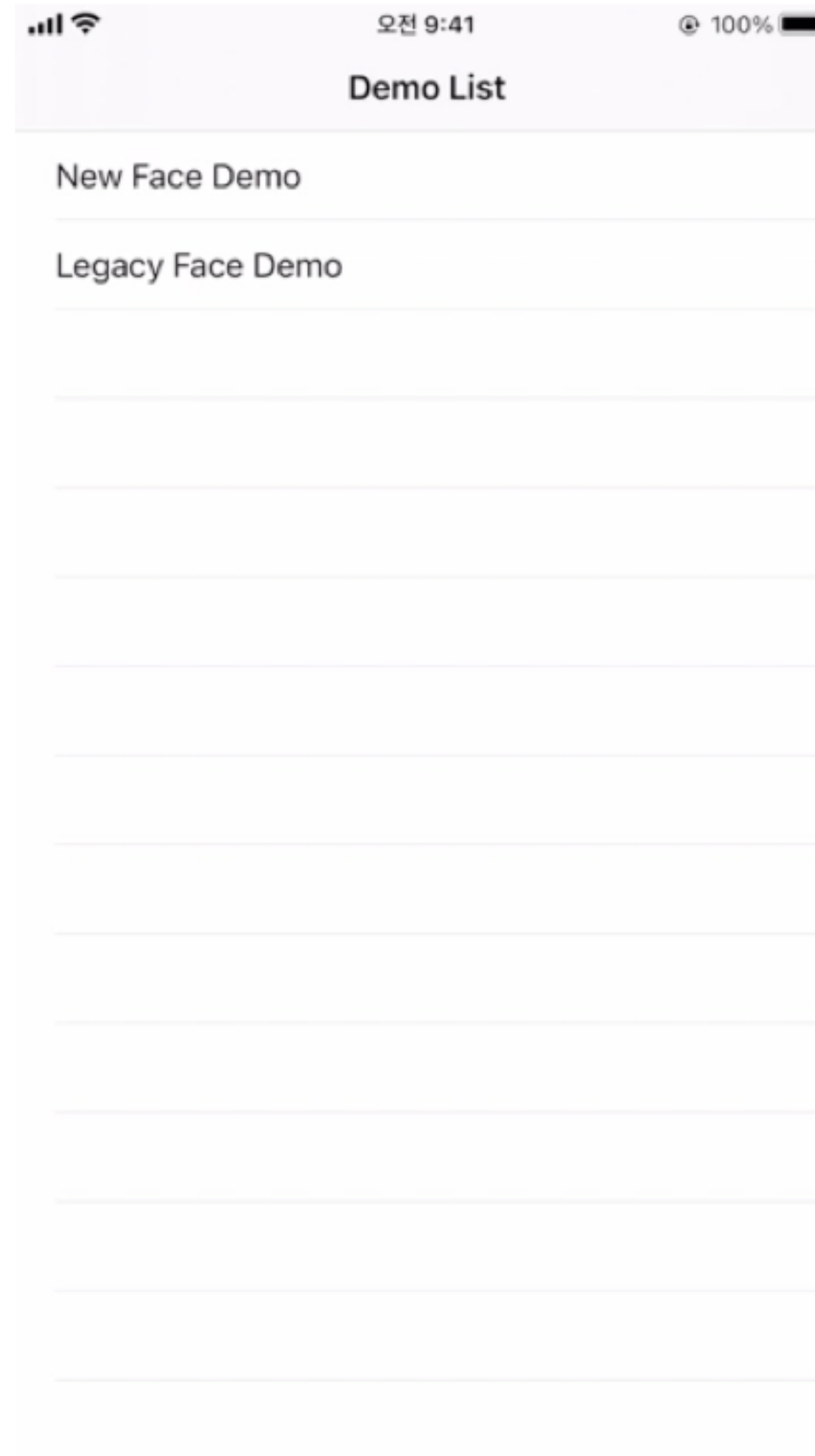
```
17
18 extension ClovaSeeWrapper {
19     func detectFace(pixelBuffer: CVPixelBuffer, informationType: CSStageInformationType, completionHandler: @escaping
20     (Result<CSFaceResult, Error>) -> Void) {
21         guard let rgbPixelBuffer = pixelBuffer.convertBGRA8888toRGB888() else {
22             completionHandler(.failure(ClovaSeeError.noPixelBuffer))
23             return
24         }
25
26         let width = CVPixelBufferGetWidth(pixelBuffer)
27         let height = CVPixelBufferGetHeight(pixelBuffer)
28         CVPixelBufferLockBaseAddress(rgbPixelBuffer, CVPixelBufferLockFlags(rawValue: 0))
29         defer {
30             CVPixelBufferUnlockBaseAddress(rgbPixelBuffer, CVPixelBufferLockFlags(rawValue: 0))
31         }
32
33         guard let rgbDataPointer = CVPixelBufferGetBaseAddress(rgbPixelBuffer) else {
34             completionHandler(.failure(ClovaSeeError.noPixelBuffer))
35             return
36         }
37
38         let frame = CSFrame(pixels: rgbDataPointer,
39                             width: CGFloat(width),
40                             height: CGFloat(height),
41                             type: CSFrameFormatType.RGB888)
42
43         // TODO(@youngsoo.lee) : expose options to the UI
44         let options = CSFaceOptionsBuilder()
45         options.informationType = informationType.rawValue
46         options.minimumBoundingBoxSize = 0.1
47         let result = self.runForFace(with: frame, options: CSFaceOptions(builder: options))
48
49         completionHandler(.success(result))
50     }
51 }
52
```

clova_see는 RGB만 지원하므로,
RGB가 아닌 경우 RGB로 변환하는 과정이 필요함

2.2. iOS Hands-On

```
47     self.presentEulerAngle(face: face, size: size)
48     self.presentMaskDetectionResult(face: face, size: size)
49     self.presentSpoofingDetectionResult(face: face, size: size)
50
51     if let capturedFace = capturedFace {
52         self.view?.drawSimilarity(similarity: CSFace.getCosineSimilarity(withFace1: face, face2:
53                                     capturedFace))
54     }
55 }
56 }
57
58 private func presentContour(face: CSFace, size: CGSize) {
59     guard let typeCasted = face.contour.points as? [NSValue] else { return }
60     let points = typeCasted.map { $0.cgPointValue }
61     self.view?.drawContour(points: points, size: size)
62 }
63
64 private func presentEulerAngle(face: CSFace, size: CGSize) {
65     let description = String(format: "[x: %4.2f, y: %4.2f, z: %4.2f]",
66                               arguments: [face.eulerAngle.x,
67                                             face.eulerAngle.y,
68                                             face.eulerAngle.z])
69
70     self.view?.drawEulerAngle(eulerAngleDescription: description,
71                               faceBoundingBoxRect: face.boundingBox.rect,
72                               size: size)
73 }
74
75 private func presentMaskDetectionResult(face: CSFace, size: CGSize) {
76     self.view?.drawMaskDetectionResult(isMasked: face.mask, faceBoundingBoxRect: face.boundingBox.rect, size: size)
77 }
78
79 private func presentSpoofingDetectionResult(face: CSFace, size: CGSize) {
80     self.view?.drawSpoofingDetectionResult(isSpoofed: face.spoof, faceBoundingBoxRect: face.boundingBox.rect, size:
81     size)
81 }
```

2.2. iOS Hands-On



2.3. Android Hands-On

2.3.1 clova-face-kit download

naver / clova-face-kit Public

Notifications Star 22 Fork 1

Code Issues Pull requests Actions Wiki Security Insights

main 1 branch 1 tag Go to file Code

정상업 and YoungjaeKim modify README.md contents in examples/ios 379e513 7 days ago 3 commits

docs	update iOS Demo & quick_developer_guide_ios_ko.md	7 days ago
examples	modify README.md contents in examples/ios	7 days ago
.gitignore	Initial commit	last month
LICENSE	Initial commit	last month
NOTICE	Initial commit	last month
README.md	Initial commit	last month

README.md

CLOVA Face Kit

👉 [Go to Release Page](#) to download the artifacts.

Introduction

CLOVA Face Kit (also known as CLOVA SEE) is an easy-to-use vision analysis SDK. The key features are as below;

- On Device AI solution
- Support cross-platform (Android, iOS, Linux, macOS, Windows)

About
On-device lightweight face recognition. Available on Android, iOS, WASM, Python.

Readme

Releases 1

0.2130 Latest
on 17 Sep

Packages
No packages published

<https://github.com/naver/clova-face-kit>

2.3.1 clova-face-kit download

Latest release

0.2130
175385f

Compare ▾

0.2130
junhee-yoo released this on 17 Sep

Released

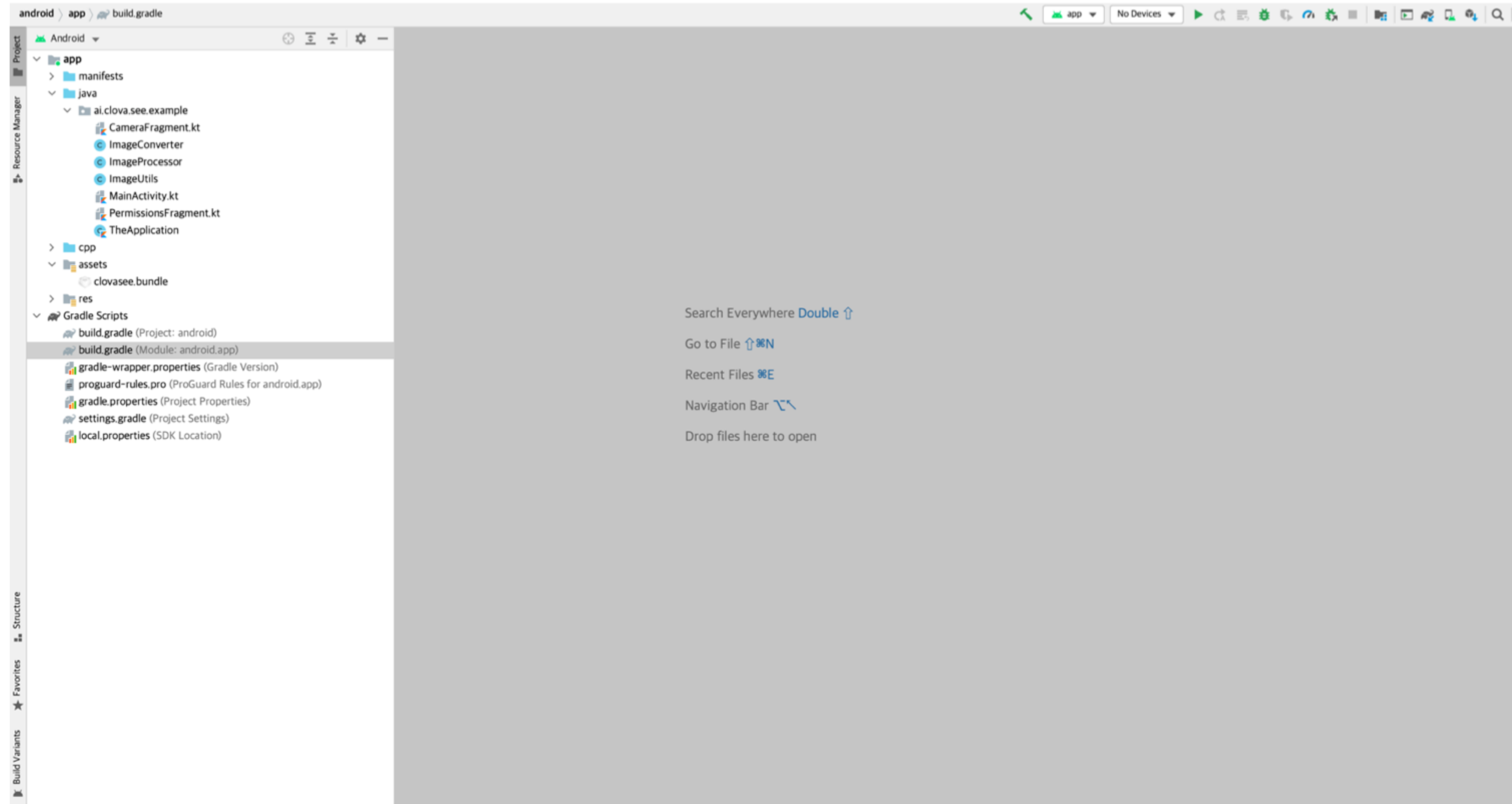
Assets 18

clova-see-0.2130.54.aar	30.2 MB
clovasee-0.2130.54-cp38-cp38-win_amd64.whl	20 MB
clovasee-0.2130.60-cp36-cp36m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp37-cp37m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp38-cp38-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-Linux.sh	3.32 MB
clovasee-0.2130.70-cp38-cp38-macosx_10_14_x86_64.whl	20.6 MB
clovasee-0.2130.70-cp39-cp39-macosx_10_15_x86_64.whl	20.6 MB
clovasee-0.2130.70-Darwin.sh	622 KB
clovasee-0.2930.0-webassembly.zip	19.6 MB
clovasee.all.bundle	20.8 MB
clovasee.body.bundle	188 KB
clovasee.face.bundle	22 MB
clovasee.face_without_spoofing_detection.bundle	6.6 MB
clovasee.ocr.bundle	3.3 MB
clova_see.xcframework-0.2130.45.zip	3.84 MB
Source code (zip)	
Source code (tar.gz)	

1) aar 파일을 다운로드

2) 사용하려는 타입의 bundle file 다운로드

2.3.2 clova-face-kit open



2.3.3 clova-face-kit source code

Project Structure: android -> app -> libs

Annotation: Project/app/libs 폴더에서 import 하도록 변경

```

60
61 dependencies {
62     implementation fileTree(dir: "libs", include: ["*.aar"])
63     // implementation files("../..../platforms/android/clova-see/build/outputs/aar/clova-see.aar")
64
65     implementation "androidx.appcompat:appcompat:1.0.2"
66     implementation "androidx.camera:camera-core:1.0.0-alpha04"
67     implementation "androidx.camera:camera-camera2:1.0.0-alpha04"
68     implementation "androidx.constraintlayout:constraintlayout:1.1.3"
69     implementation "androidx.core:core-ktx:1.0.2"
70     implementation "androidx.navigation:navigation-fragment-ktx:2.0.0"
71     implementation "androidx.navigation:navigation-ui-ktx:2.0.0"
72     implementation "com.jakewharton.timber:timber:4.7.1"
73     implementation "io.reactivex.rxjava2:rxkotlin:2.4.0"
74     implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk8:$kotlin_version"
75 }
76
dependencies{}

```

2.3.3 clova-face-kit source code

```

ImageConverter.java
15
16 package ai.clova.see.example;
17
18 import android.graphics.Bitmap;
19
20 public class ImageConverter {
21     static {
22         System.loadLibrary("image_converter");
23     }
24
25     public native Bitmap nv21ToARGB(byte[] rawData, int width, int height);
26
27     public native Bitmap rotateImage(byte[] rawData, int width, int height, int rotationDegrees);
28
29 }
30

ImageUtils.java
15
16 package ai.clova.see.example;
17
18 import ...
19
20 /** Utility class for manipulating images. */
21 public class ImageUtils {
22     // This value is 2 ^ 18 - 1, and is used to clamp the RGB values
23     // are normalized to eight bits.
24     static final int kMaxChannelValue = 262143;
25
26     @SuppressWarnings("unused")
27     private static final Logger LOGGER = new Logger();
28
29     /**
30      * Utility method to compute the allocated size in bytes of a
31      * dimensions.
32      */
33     public static int getYUVByteSize(final int width, final int height) {
34         // The luminance plane requires 1 byte per pixel.
35         final int ySize = width * height;
36
37         // The UV plane works on 2x2 blocks, so dimensions with
38         // Each 2x2 block takes 2 bytes to encode, one each for U
39         final int uvSize = ((width + 1) / 2) * ((height + 1) / 2) * 2;
40
41         return ySize + uvSize;
42     }
43
44     /**
45      * Saves a Bitmap object to disk for analysis.
46      *
47      * @param bitmap The bitmap to save.
48      */
49     public static void saveBitmap(final Bitmap bitmap) { saveBitmap(bitmap, "image.png"); }
50
51     /**
52      * Saves a Bitmap object to disk for analysis.
53      *
54      * @param bitmap The bitmap to save.
55      * @param filename The location to save the bitmap to.
56     }
57
58 ImageProcessor.java
59 package ai.clova.see.example;
60
61 import ...
62
63 public class ImageProcessor {
64     @NonNull
65     private CameraX.LensFacing lensFacing;
66     private ImageConverter converter = new ImageConverter();
67
68     public ImageProcessor(@NonNull CameraX.LensFacing lensFacing) {
69         this.lensFacing = lensFacing;
70     }
71
72     @Nullable
73     public Bitmap toBitmap(@Nullable Image image, int rotationDegrees) {
74         if (image == null || image.getWidth() == 0 || image.getHeight() == 0) {
75             return null;
76         }
77
78         final int imageWidth = image.getWidth();
79         final int imageHeight = image.getHeight();
80
81         // YUV420_888 포맷을 NV21 포맷의 데이터 형태로 변환합니다.
82         byte[] data = yuv420ToNV21(image);
83         // NV21포맷의 ByteArray를 ARGB8888 포맷으로 변환합니다.
84         Bitmap rgbBitmap = converter.nv21ToARGB(data, imageWidth, imageHeight);
85         // ARGB8888로 변환된 비트맵을 Rotation 및 Flip을 수행합니다.
86         final float sx = lensFacing == CameraX.LensFacing.FRONT ? 1 : -1;
87         final float sy = lensFacing == CameraX.LensFacing.FRONT ? 1 : -1;
88         Bitmap rotateBitmap = toRotateBitmap(rgbBitmap, rotationDegrees, sx, sy);
89
90         return rotateBitmap;
91     }
92
93     private Bitmap toRotateBitmap(Bitmap bitmap,
94                                 final int rotationDegrees,
95                                 final float sx,
96                                 final float sy) {
97         int rotationType = toRotationType(rotationDegrees, sx);
98         // rotationType이 1인 경우는 bitmap을 그대로 반환 해 줍니다.
99         if (rotationType == 1) {
100             return bitmap;
101         }
102
103         // 90, 180, 270 또는 scale이 1.0이 아닐 경우에는 기존의 로직을 그대로
    
```

CameraX에서 받은 ImageProxy를 bitmap으로 변환

2.3.3 clova-face-kit source code

```

1 //...
19
20 #if defined(__ARM_NEON__) || defined(__ARM_NEON) || defined(_M_ARM) || defined(_M_ARM64) || defined(_M_HYBRID_X86_ARM64)
21 ...
22 #endif
23
24 #include "yuv2rgb.h"
25
26 #define align(v, a) ((v) + ((a) - 1) & ~((a) - 1))
27
28 // BT.709 - Video Range
29 // Y U V
30 // R = 1.164384 0.000000 1.792741
31 // G = 1.164384 -0.213249 -0.532909
32 // B = 1.164384 2.112402 0.000000
33 //
34 // BT.709 - Full Range
35 // Y U V
36 // R = 1.000000 0.000000 1.581000
37 // G = 1.000000 -0.188062 -0.469967
38 // B = 1.000000 1.862906 0.000000
39
40 #define vY 1.164384
41 #define vU -0.213249
42 #define vUB 2.112402
43 #define vVR 1.792741
44 #define vVG -0.532909
45 #define vY 1.000000
46 #define vUG -0.188062
47 #define vUB 1.862906
48 #define vVR 1.581000
49 #define vVG -0.469967
50
51 <int rgb_width,bool rgb_swizzle,bool interleaved,bool first_u,bool full_range>
52 void YUV2RGB(
53     int width, int height,
54     const void *y, const void *u, const void *v,
55     int stride_y, int stride_u, int stride_v,
56     void *rgb, int stride_rgb) {
57     if (stride_rgb < 0) {
58         rgb = static_cast<char *>(rgb) - (stride_rgb * (height - 1));
59     }

```

yuv_420_888 ->
RGB로 변환 하는
코드는 C++로 작성

2.3.3 clova-face-kit source code

Camera Fragment

```

311  */
312  private inner class ClovaSeeRunner(
313      context: Context,
314      lensFacing: CameraX.LensFacing,
315      listener: ClovaSeeRunnerListener? = null
316  ) : ImageAnalysis.Analyzer {
317      private val listeners =
318          ArrayList<ClovaSeeRunnerListener>().apply { listener?.let { add(it) } }
319      private val imageProcessor = ImageProcessor(lensFacing)
320      private val clovaSee: ClovaSee
321
322      private val frameRateWindow = 8
323      private val frameTimestamps = ArrayDeque<Long>( numElements= 5)
324      var framesPerSecond: Float = -1.0f
325      private set
326
327      init {
328          val settings = SettingsBuilder()
329              .setIntermittentInformationRatio(1)
330              .setNumberOfThreads(4) // Set the maximum number of threads to be used by ClovaSee.
331              .setPerformanceMode(Settings.PerformanceMode.ACCURATE_106)
332              .build()
333          clovaSee = ClovaSee(context, settings)
334          // clovaSee = ClovaSee(context, settings, "file:///android_asset/clovasee.bundle")
335          // clovaSee = ClovaSee(context, settings, "/data/local/tmp/clovasee.bundle")
336      }
337
338      override fun analyze(imageProxy: ImageProxy?, rotationDegrees: Int) {
339          if (listeners.isEmpty())
340              return
341
342          // convertYUV420ToARGB8888 is very naive implementation!!
343          // It is recommended to use the optimized implementation within the service!!
344          val bitmap = imageProcessor.toBitmap(imageProxy!!.image, rotationDegrees)
345
346          // val testInput = BitmapFactory.decodeResource(resources, R.drawable.test_input_body)
347          // val testInput = BitmapFactory.decodeResource(resources, R.drawable.test_input_face)
348          // val testInput = BitmapFactory.decodeResource(resources, R.drawable.test_input_ocr)
349          // val bitmap = Bitmap.createBitmap(testInput.width, testInput.height, testInput.config)
350          // val canvas = Canvas(bitmap)
351          // canvas.drawBitmap(testInput, 0.0f, 0.0f, Paint())
352

```

2.3.3 clova-face-kit source code

```
init {  
    val settings = SettingsBuilder()  
        .setIntermittentInformationRatio(1)  
        .setNumberOfThreads(4) // Set the maximum number of threads to be used by ClovaSee.  
        .setPerformanceMode(Settings.PerformanceMode.ACCURATE_106)  
        .build()  
    clovaSee = ClovaSee(context, settings)  
    // clovaSee = ClovaSee(context, settings, "file:///android_asset/clovasee.bundle")  
    // clovaSee = ClovaSee(context, settings, "/data/local/tmp/clovasee.bundle")  
}
```

- assets
 - clovasee.bundle

2.3.3 clova-face-kit source code

```
/**  
 * ClovaSeeController  
 */  
interface ClovaSeeController {  
    enum class RunType { FACE, BODY, OCR }  
  
    fun isBypassed(): Boolean  
    fun setBypassed(set: Boolean)  
    fun getRunType(): RunType  
    fun setRunType(set: RunType)  
    fun isLogging(): Boolean  
    fun setIsLogging(isLogging: Boolean)  
}
```

FACE, BODY, OCR

총 3가지 타입이 있어요

2.3.3 clova-face-kit source code

```

ClovaSeeController.RunType.BODY -> {
    Pair(
        ClovaSeeRunResult(
            ai.clova.see.face.Result(emptyArray()),
            clovaSee.run(bitmap, ai.clova.see.body.OptionsBuilder().build()),
            ai.clova.see.ocr.Result(Document())
        ),
        clovaSee.getMeasureResult()
    )
}

ClovaSeeController.RunType.OCR -> {
    Pair(
        ClovaSeeRunResult(
            ai.clova.see.face.Result(emptyArray()),
            ai.clova.see.body.Result(Segment( size: 0)),
            clovaSee.run(bitmap, ai.clova.see.ocr.OptionsBuilder().build())
        ),
        clovaSee.getMeasureResult()
    )
}

else -> {
    val faceOptions = ai.clova.see.face.OptionsBuilder()
        .setBoundingBoxThreshold(0.7f)
        .setInformationToObtain(ai.clova.see.face.Options.CONTOURS or
            ai.clova.see.face.Options.MASKS or
            ai.clova.see.face.Options.EULER_ANGLES or
            ai.clova.see.face.Options.TRACKING_IDS)
        .setResizeThreshold(320)
        .setMinimumBoundingBoxSize(0.1f)
        .build()
    Pair(
        ClovaSeeRunResult(
            clovaSee.run(bitmap, faceOptions),
            ai.clova.see.body.Result(Segment( size: 0)),
            ai.clova.see.ocr.Result(Document())
        ),
        clovaSee.getMeasureResult()
    )
}

```

사용하려는 type의 option을 설정해서 ClovaSee.run() 을 실행

2.3.3 clova-face-kit source code

```
val faceOptions = ai.clova.see.face.OptionsBuilder()
    .setBoundingBoxThreshold(0.7f)
    .setInformationToObtain(ai.clova.see.face.Options.CONTOURS or
        ai.clova.see.face.Options.MASKS or
        ai.clova.see.face.Options.EULER_ANGLES or
        ai.clova.see.face.Options.TRACKING_IDS)
    .setResizeThreshold(320)
    .setMinimumBoundingBoxSize(0.1f)
    .build()
Pair(
    ClovaSeeRunResult(
        clovaSee.run(bitmap, faceOptions),
        ai.clova.see.body.Result(Segment(size: 0)),
        ai.clova.see.ocr.Result(Document())
    ),
    clovaSee.getMeasureResult()
)
```

2.3.3 clova-face-kit source code

```
public final class Options public constructor(boundingBoxThreshold: kotlin.Float /* = compiled code
public companion object {
    public final val ALL: kotlin.Int /* compiled code */

    public final val BOUNDING_BOXES: kotlin.Int /* compiled code */

    public final val CONTOURS: kotlin.Int /* compiled code */

    public final val DEFAULT_BOUNDING_BOX_THRESHOLD: kotlin.Float /* compiled code */

    public final val DEFAULT_MINIMUM_BOUNDING_BOX_SIZE: kotlin.Float /* compiled code */

    public final val DEFAULT_RESIZE_THRESHOLD: kotlin.Int /* compiled code */

    public final val EULER_ANGLES: kotlin.Int /* compiled code */

    public final val FEATURES: kotlin.Int /* compiled code */

    public final val MASKS: kotlin.Int /* compiled code */

    public final val MOJOS: kotlin.Int /* compiled code */

    public final val SPOOFS: kotlin.Int /* compiled code */
```

2.3.3 clova-face-kit source code

```
public final class Face public constructor(boundingBox: android.graphics.Rect, contour: ai.clova.see
    public companion object {
        @kotlin.jvm.JvmStatic public final fun getCosineSimilarity(face1: ai.clova.see.Face, face2:
        @kotlin.jvm.JvmStatic public final fun isSame(face1: ai.clova.see.Face, face2: ai.clova.see.
    }

    public final val boundingBox: android.graphics.Rect /* compiled code */

    public final val contour: ai.clova.see.Contour /* compiled code */

    public final val eulerAngle: ai.clova.see.EulerAngle /* compiled code */

    public final val feature: ai.clova.see.Feature /* = kotlin.FloatArray */ /* compiled code */

    public final val mask: ai.clova.see.Mask /* = kotlin.Boolean */ /* compiled code */

    public final val mojo: ai.clova.see.Mojo /* = kotlin.ByteArray */ /* compiled code */

    public final val spoof: ai.clova.see.Spoof /* = kotlin.Boolean */ /* compiled code */

    public final val trackingID: ai.clova.see.TrackingID /* = kotlin.Int */ /* compiled code */
}
```

Face 결과 값

얼굴과 관련된 다양한
정보를 return 값으로
받을 수 있다

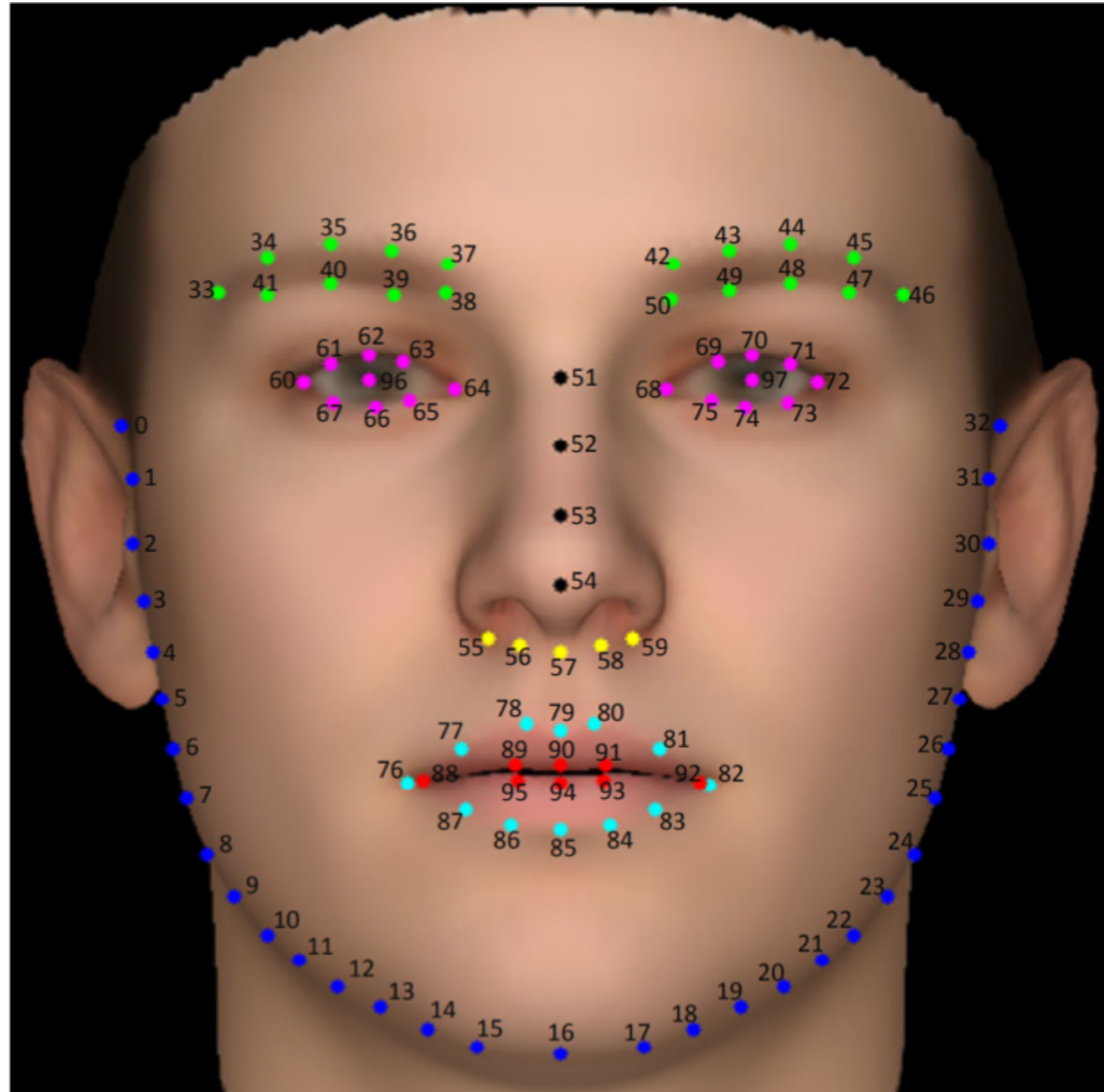
2.3.4 clova-face-kit 실행

The screenshot displays an IDE environment for an Android project. On the left, the Project Manager shows the directory structure: `android > app > src > main > java > ai > clova > see > example`. The file `build.gradle (:app)` is selected, with line 403 highlighted in yellow. A dark overlay in the center shows the Clova Face Kit logo and the GitHub URL `https://github.com/naver/clova-face-kit`. The right side of the IDE shows a code editor with a yellow highlight and a cursor. At the bottom, a Performance monitor window is open, displaying the following data:

Performance		27.89fps
Native Total		115.11fps
Detector		7.62ms
Tracker		0.00ms
Aligner		0.00ms
Landmarker		0.00ms
Estimator		0.00ms
Recognizer		0.00ms
Mask Detector		0.00ms
Spoofer Detector		0.00ms

The IDE status bar at the bottom shows the time `403:18`, encoding `CRLF`, `UTF-8`, and `4 spaces`. A message at the bottom left states: `Failed to start monitoring R54M80ONCLJ (5 minutes ago)`.

2.3.5 clova-face-kit 응용



왼쪽 기준

눈 윗줄 (61~63) 과

눈 아랫줄 (65~67) 사이의 거리를

계산해서 눈 뜨고있는지 여부 계산

2.3.5 clova-face-kit 응용

```

/**
 * 왼쪽 눈을 뜨고 있는지 체크
 */
fun getLeftEyeOpen(face: Face): Double {
    face.contour.points.let { it: Points
        return (distance(
            it[61],
            it[67]
        ) + distance(
            it[62],
            it[66]
        ) + distance(
            it[63],
            it[65]
        ))
    }
}

```

```

/**
 * 오른쪽 눈을 뜨고 있는지 체크
 */
fun getRightEyeOpen(face: Face): Double {
    face.contour.points.let { it: Points
        return (distance(
            it[69],
            it[75]
        ) + distance(
            it[70],
            it[74]
        ) + distance(
            it[71],
            it[73]
        ))
    }
}

```

2.3.5 clova-face-kit 응용

```
public final class EulerAngle public constructor(x: kotlin.Float, y: kotlin.Float, z: kotlin.Float) {  
    public final val pitch: kotlin.Float /* compiled code */  
  
    public final val roll: kotlin.Float /* compiled code */  
  
    public final val x: kotlin.Float /* compiled code */  
  
    public final val y: kotlin.Float /* compiled code */  
  
    public final val yaw: kotlin.Float /* compiled code */  
  
    public final val z: kotlin.Float /* compiled code */  
}
```

EulerAngle을 통해 얼굴의 방향을 알 수 있어요

2.3.5 clova-face-kit 응용

```
/**
 * 얼굴 각도 체크
 */
enum class FacePose {
    FRONT,
    DOWN,
    UP,
    ROLL_LEFT,
    ROLL_RIGHT,
    PAN_RIGHT,
    PAN_LEFT
}
```

```
fun getFacePose(
    angle: Triple<Float, Float, Float>,
): String {
    angle.let { (pitch, roll, yaw) ->
        var value = pitch
        var valueThreshold = 8.0f
        val rollThreshold = 12.0f
        val yawThreshold = 12.0f
        var facePoseValueIndex = FacePose.DOWN.ordinal

        if (getPoseRatio(value, valueThreshold) < getPoseRatio(roll, rollThreshold)) {
            value = roll
            valueThreshold = rollThreshold
            facePoseValueIndex = FacePose.ROLL_RIGHT.ordinal
        }
        if (getPoseRatio(value, valueThreshold) < getPoseRatio(yaw, yawThreshold)) {
            value = yaw
            valueThreshold = yawThreshold
            facePoseValueIndex = FacePose.PAN_LEFT.ordinal
        }

        return if (getPoseRatio(value, valueThreshold) < 1.0) {
            FacePose.FRONT.name
        } else {
            FacePose.getFromValue(if (value < 0) facePoseValueIndex else facePoseValueIndex+1).name
        }
    }
}
```


2.3.6 clova-face-kit 응용 코드 실행

The screenshot shows an IDE environment with the following components:

- Project Manager:** Shows the project structure for an Android app, including folders for manifests, java, assets, and res. The 'clovasee.bundle' folder is highlighted.
- Code Editor:** Displays Kotlin code for 'CameraFragment.kt'. The code includes imports for 'Face' and 'EulerAngle', and a function 'getFaceEulerAngle' that returns a list of 'Face' objects. The code is partially obscured by a dark overlay.
- Gradle Console:** Shows the output of the build process, including the path 'android.app:android.app:res:clovasee.bundle'.
- Emulator:** A window titled 'Clova Face Kit' displays the results of the face detection process. The output is:

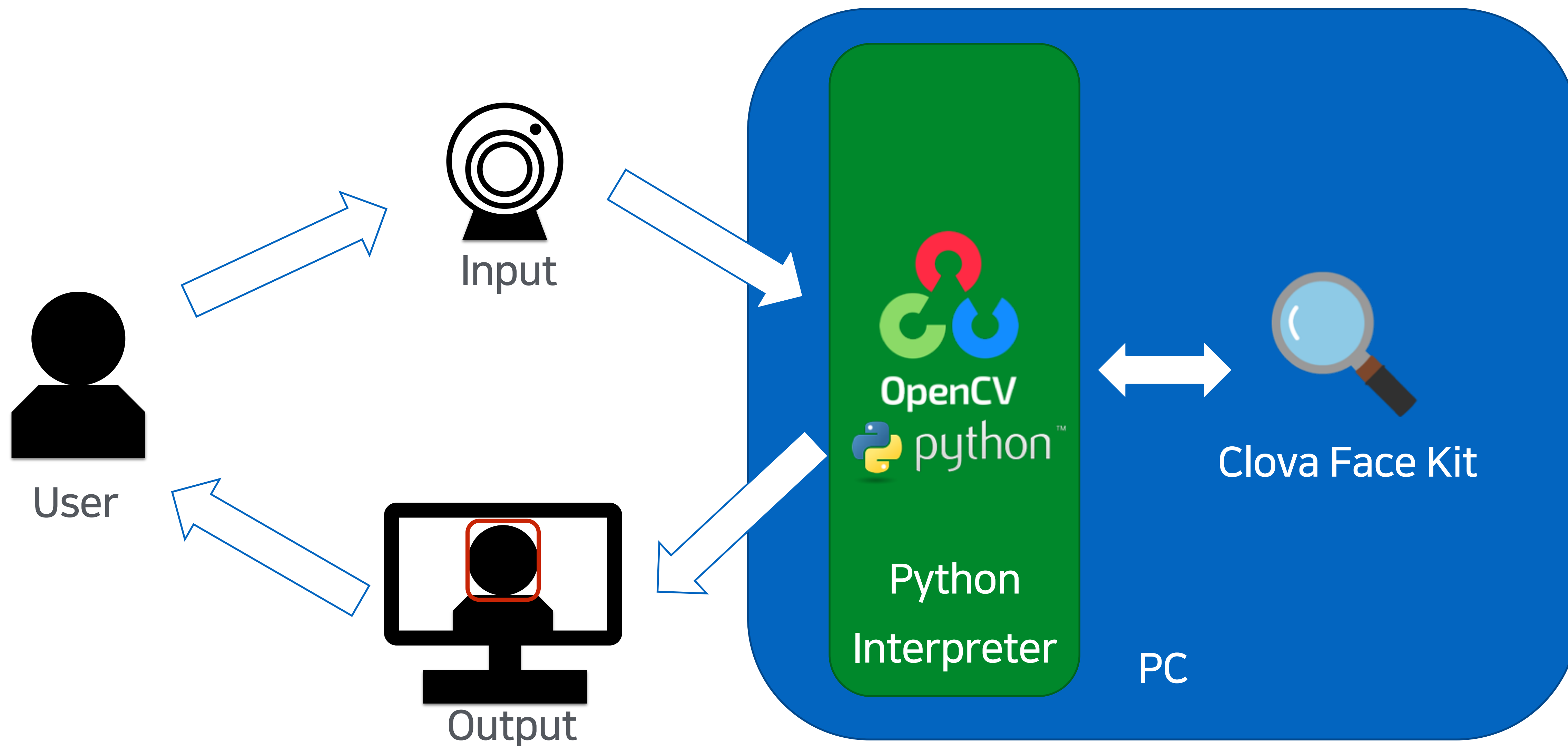

```
Left Eye : null
Right Eye : null
Face Euler Angle : null
```

The IDE interface also shows various toolbars and a status bar at the bottom with information like '689:38 CRLF UTF-8 4 spaces'.

2.4. Python Hands-On



2.4. Python Hands-On



2.4. Python Hands-On

Preview



2.4. Python Hands-On

Download

naver / clova-face-kit Public

Unwatch 9 Unstar 22 Fork 1

Code Issues Pull requests ZenHub Actions Wiki Security Insights

Releases Tags Draft a new release

Latest release

0.2130
175385f

Compare

0.2130 Edit

junhee-yoo released this on Sep 17

Released

Assets 18

clova-see-0.2130.54.aar	30.2 MB
clovasee-0.2130.54-cp38-cp38-win_amd64.whl	20 MB
clovasee-0.2130.60-cp36-cp36m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp37-cp37m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp38-cp38-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-Linux.sh	3.32 MB
clovasee-0.2130.70-cp38-cp38-macosx_10_14_x86_64.whl	20.6 MB
clovasee-0.2130.70-cp39-cp39-macosx_10_15_x86_64.whl	20.6 MB
clovasee-0.2130.70-Darwin.sh	622 KB
clovasee-0.2930.0-webassembly.zip	19.6 MB

2.4. Python Hands-On

Download



https://commons.wikimedia.org/wiki/File:Guy_fawkes_mask_by_nacreouss-d462juf.png

-> mask.png

2.4. Python Hands-On

Environments

```

→ devview_2021 pipenv shell
Creating a virtualenv for this project...
Pipfile: /Users/jhyoo/workspace/area51/devview_2021/Pipfile
Using /usr/local/bin/python3.9 (3.9.6) to create virtualenv...
:: Creating virtual environment...created virtual environment CPython3.9.6.final.0-64 in 598m
s
  creator CPython3Posix(dest=/Users/jhyoo/.local/share/virtualenvs/devview_2021-FYBE5rkn, cle
ar=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy,
app_data_dir=/Users/jhyoo/Library/Application Support/virtualenv)
  added seed packages: clovasee==0.2130.70, numpy==1.21.2, opencv_python==4.5.3.56, pip==2
1.2.4, pybind11==2.7.1, setuptools==57.4.0, setuptools==58.0.4, wheel==0.37.0
  activators BashActivator,CShellActivator,FishActivator,PowerShellActivator,PythonActivator
,XonshActivator

✓ Successfully created virtual environment!
Virtualenv location: /Users/jhyoo/.local/share/virtualenvs/devview_2021-FYBE5rkn
Launching subshell in virtual environment...
. /Users/jhyoo/.local/share/virtualenvs/devview_2021-FYBE5rkn/bin/activate
→ devview_2021 . /Users/jhyoo/.local/share/virtualenvs/devview_2021-FYBE5rkn/bin/activate
(devview_2021) → devview_2021 pip install opencv-python

```

2.4. Python Hands-On

Imports

```
import cv2

from clovasee import ClovaSee
from clovasee import Frame
from clovasee import Settings
from clovasee import SettingsBuilder
from clovasee import FaceOptions
from clovasee import FaceOptionsBuilder
```


2.4. Python Hands-On

Main function

(setup)

```
def main():
    capture = cv2.VideoCapture(1)
    capture.set(cv2.CAP_PROP_FORMAT, cv2.CV_8UC3)

    settings = SettingsBuilder() \
        .set_number_of_threads(4) \
        .set_performance_mode(Settings.PerformanceMode.kAccurate106) \
        .build()
    clova_see = ClovaSee(settings)

    options = FaceOptionsBuilder() \
        .set_bounding_box_threshold(0.7) \
        .set_information_to_obtain(
            FaceOptions.kBoundingBoxes
            | FaceOptions.kContours \
            | FaceOptions.kEulerAngles \
        ) \
        .set_minimum_bounding_box_size(0.1) \
        .set_resize_threshold(320) \
        .build()

    mask = cv2.imread('mask.png', cv2.IMREAD_UNCHANGED)
```

2.4. Python Hands-On

Main function

(loop)

```
while True:
    ret, snapshot = capture.read()
    if not ret:
        continue

    snapshot = cv2.flip(snapshot, 1)

    (height, width, _) = snapshot.shape
    frame = Frame(snapshot,
                  width,
                  height,
                  Frame.Format.kBGR_888)
    faces = clova_see.run(frame, options).faces()

    for face in faces:
        draw_bounding_box(snapshot, face)
        draw_mask(snapshot, face, mask)
        draw_contour(snapshot, face)
        draw_euler_angle(snapshot, face)

    cv2.imshow("devview 2021", snapshot)

    key = cv2.waitKey(1)
    if key == ord('q'):
        break

capture.release()
cv2.destroyAllWindows()
```

2.4. Python Hands-On

Draw functions

(bounding box, euler angle, contour)

```
def draw_bounding_box(canvas, face):
    cv2.rectangle(canvas,
                  (face.bounding_box().origin().x,
                   face.bounding_box().origin().y),
                  (face.bounding_box().right_bottom().x,
                   face.bounding_box().right_bottom().y),
                  (0, 0, 255), 2)

def draw_euler_angle(canvas, face):
    cv2.putText(canvas,
                (f"x={face.euler_angle().x:.2f} "
                 f"y={face.euler_angle().y:.2f} "
                 f"z={face.euler_angle().z:.2f}"),
                (face.bounding_box().x, face.bounding_box().right_bottom().y+18),
                cv2.FONT_HERSHEY_SIMPLEX, 0.6, (0, 0, 255))

def draw_contour(canvas, face):
    for point in face.contour().points:
        cv2.circle(canvas, (point.x, point.y), 1, (0, 255, 0))
```

2.4. Python Hands-On

Draw functions

(mask)

```
def draw_mask(canvas, face, mask):
    box = face.bounding_box()

    y = 0 if box.origin().y < 0 else box.origin().y
    x = 0 if box.origin().x < 0 else box.origin().x

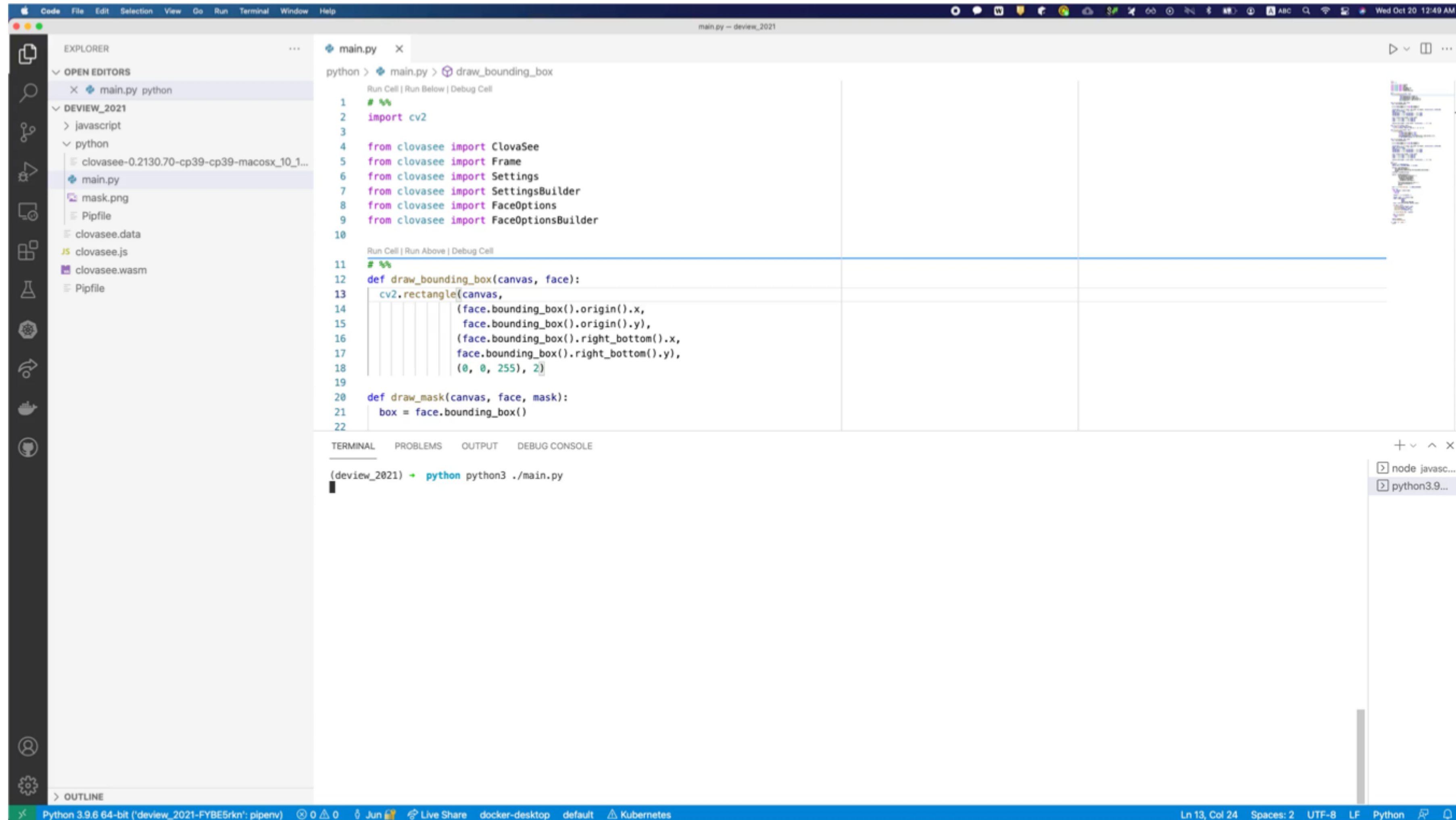
    resized_mask = cv2.resize(mask, (box.width, box.height), interpolation=cv2.INTER_AREA)
    alpha = resized_mask[:, :, 3] / 255.0
    alpha_inv = 1.0 - alpha
    resized_mask[:, :, 0] = resized_mask[:, :, 0] * alpha
    resized_mask[:, :, 1] = resized_mask[:, :, 1] * alpha
    resized_mask[:, :, 2] = resized_mask[:, :, 2] * alpha

    crop = canvas[y:y+box.height, x:x+box.width]
    crop[:, :, 0] = crop[:, :, 0] * alpha_inv
    crop[:, :, 1] = crop[:, :, 1] * alpha_inv
    crop[:, :, 2] = crop[:, :, 2] * alpha_inv

    canvas[y:y+box.height, x:x+box.width] = resized_mask[:, :, :3] + crop
```

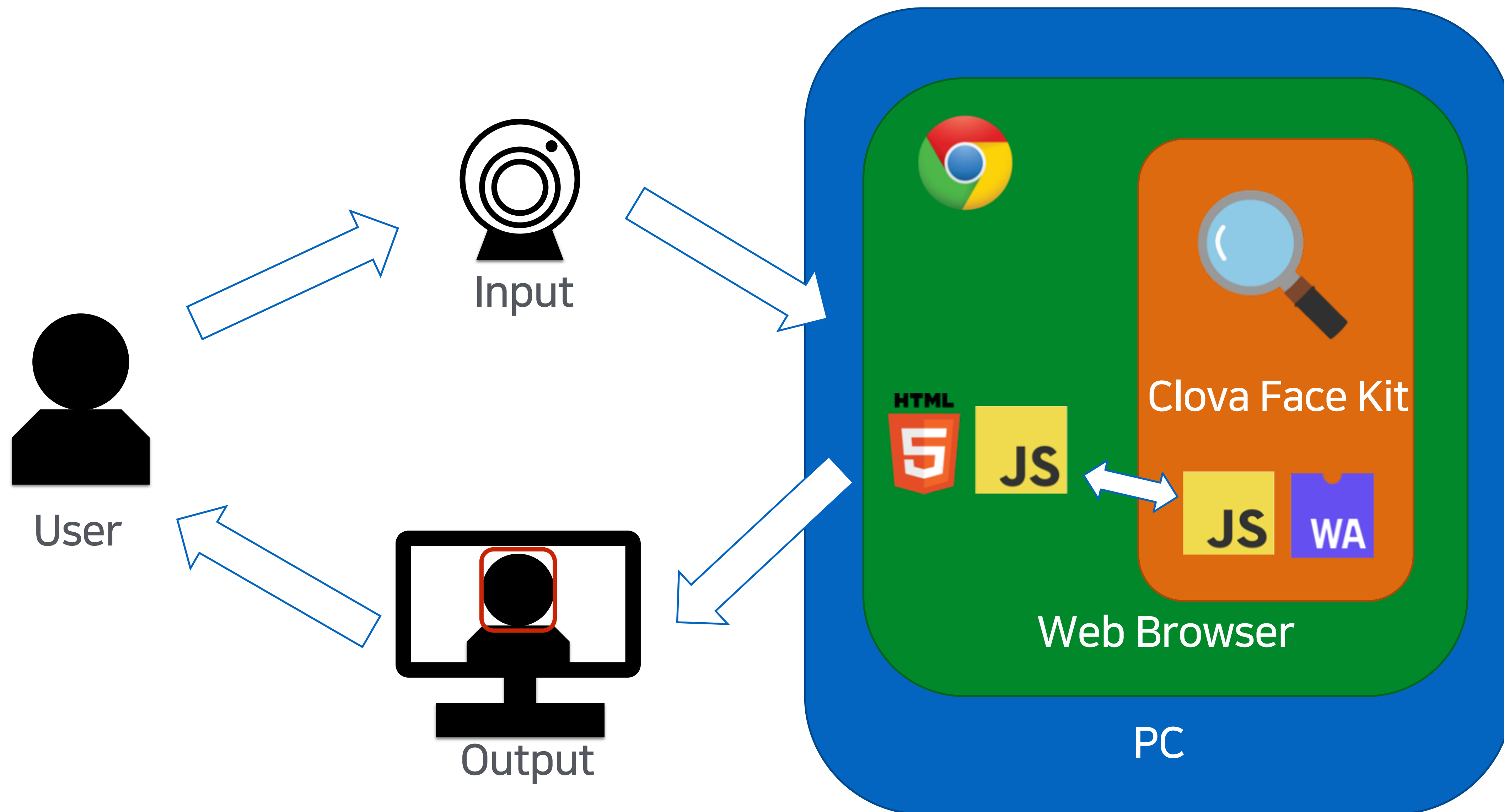
2.4. Python Hands-On

Result



2.5. JavaScript Hands-On

2.5. JavaScript Hands-On



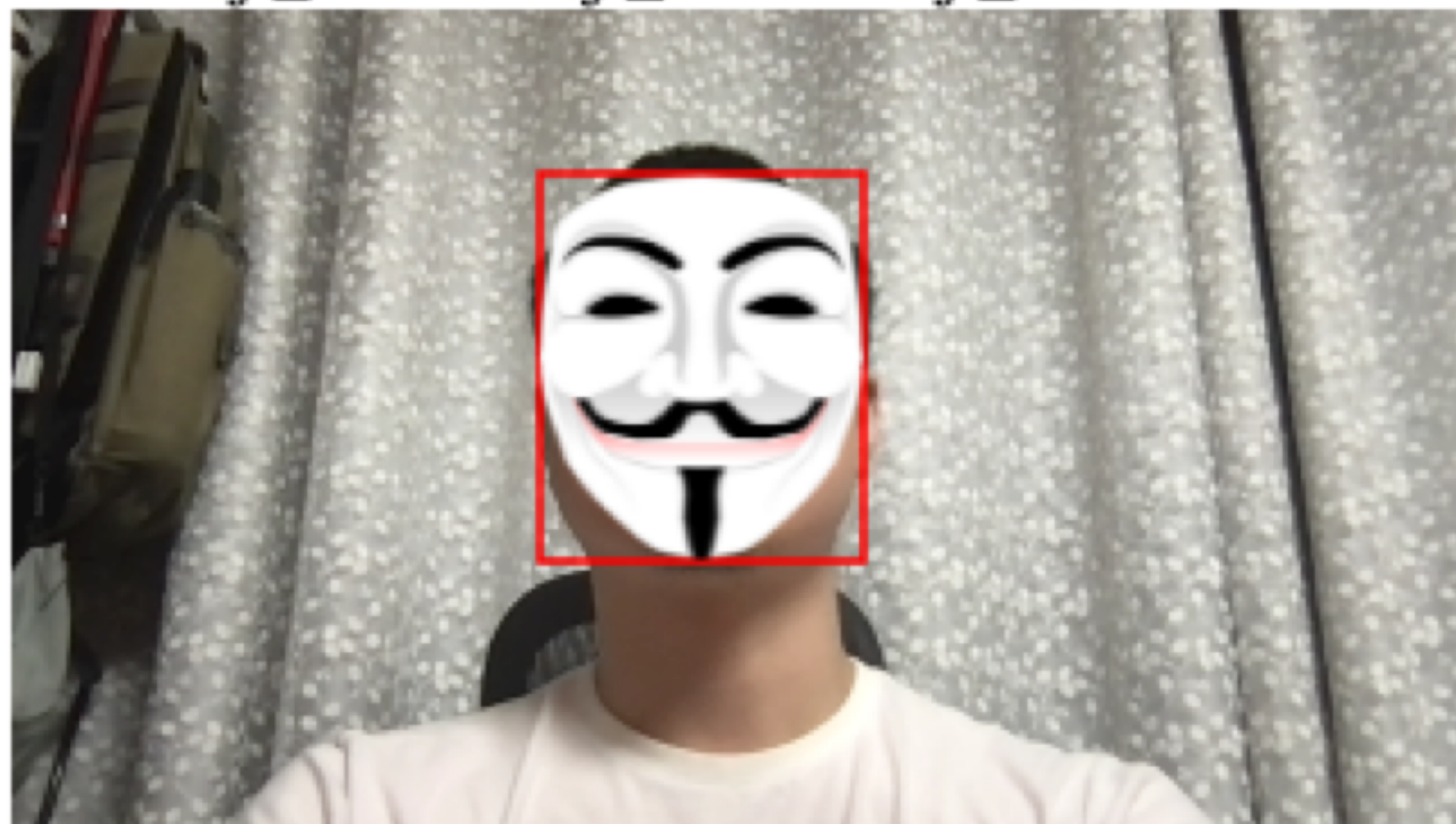
2.5. JavaScript Hands-On

Preview

Start

contour angle

emoji_0 emoji_1 emoji_2 none



2.5. JavaScript Hands-On

Download

naver / clova-face-kit Public

Unwatch 9 Unstar 22 Fork 1

Code Issues Pull requests ZenHub Actions Wiki Security Insights

Releases Tags Draft a new release

Latest release

0.2130
175385f

Compare

0.2130 Edit

junhee-yoo released this on Sep 17

Released

Assets 18

clova-see-0.2130.54.aar	30.2 MB
clovasee-0.2130.54-cp38-cp38-win_amd64.whl	20 MB
clovasee-0.2130.60-cp36-cp36m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp37-cp37m-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-cp38-cp38-linux_x86_64.whl	20.8 MB
clovasee-0.2130.60-Linux.sh	3.32 MB
clovasee-0.2130.70-cp38-cp38-macosx_10_14_x86_64.whl	20.6 MB
clovasee-0.2130.70-cp39-cp39-macosx_10_15_x86_64.whl	20.6 MB
clovasee-0.2130.70-Darwin.sh	622 KB
clovasee-0.2930.0-webassembly.zip	19.6 MB

2.5. JavaScript Hands-On

Environments

```
→ devview_2021 unzip ~/Downloads/clovasee-0.2930.0-webassembly.zip
Archive:  /Users/jhyoo/Downloads/clovasee-0.2930.0-webassembly.zip
  inflating: clovasee.data
  inflating: clovasee.js
  inflating: clovasee.wasm
→ devview_2021 npx serve .
Need to install the following packages:
  serve
Ok to proceed? (y) y
```

Serving!

```
- Local:          http://localhost:5000
- On Your Network: http://192.168.55.42:5000
```

Copied local address to clipboard!

2.5. JavaScript Hands-On

index.html

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title>Clova See for DEVVIEW 2021</title>
5  <script type="application/javascript" src="clovasee.js"></script>
6  <script type="application/javascript" src="capture.js"></script>
7  </head>
8  <body>
9  <div class="contentarea">
10 <div class="content">
11 <video id="video" hidden=true>Video stream not available.</video>
12 <button id="startbutton">Start</button>
13 <div class="features">
14 <label><input type="checkbox" id="bounding_box" name="bounding_box">bounding box</label>
15 <label><input type="checkbox" id="contour" name="contour">contour</label>
16 <label><input type="checkbox" id="angle" name="angle">angle</label>
17 </div>
18 <div class="selector">
19 <label><input type="radio" id="emoji_0" name="selector" value="0"/>emoji_0</label>
20 <label><input type="radio" id="emoji_1" name="selector" value="1"/>emoji_1</label>
21 <label><input type="radio" id="emoji_2" name="selector" value="2"/>emoji_2</label>
22 <label><input type="radio" id="emoji_3" name="selector" value="3" checked="checked"/>none</label>
23 </div>
24 </div>
25 <div class="output">
26 <canvas id="canvas" hidden=true></canvas>
27 <img id="photo" alt="The screen capture will appear in this box.">
28 </div>
29 </div>
30 </body>
31 </html>
32
```

2.5. JavaScript Hands-On

capture.js

(async load)

```
clova().then(  
  (instance) => {  
    console.log('Wasm Loaded!');  
    process(instance);  
  }  
);
```

2.5. JavaScript Hands-On

capture.js

(create instance)

```
function process(clova) {
  // The width and height of the captured photo. We will set the
  // width to the value defined here, but the height will be
  // calculated based on the aspect ratio of the input stream.

  var width = 320;    // We will scale the photo width to this
  var height = 0;    // This will be computed based on the input stream

  // |streaming| indicates whether or not we're currently streaming
  // video from the camera. Obviously, we start at false.

  var streaming = false;

  // The various HTML elements we need to configure or control. These
  // will be set by the startup() function.
  var video = document.getElementById('video');
  var canvas = document.getElementById('canvas');
  var photo = document.getElementById('photo');
  var startbutton = document.getElementById('startbutton');

  let resources = new clova.Resources('clovasee.all.bundle');
  let settings = new clova.SettingsBuilder()
    .setPerformanceMode(clova.PerformanceMode.ACCURATE106)
    .build();
  let clovaSee = new clova.ClovaSee(settings, resources);

  var masks = [
    document.createElement("img"),
    document.createElement("img"),
    document.createElement("img"),
  ]
  masks.forEach((item, idx) => {
    item.src = "./mask_" + idx + ".png";
  });
  masks.push(document.createElement("img"));
}
```

2.5. JavaScript Hands-On

capture.js

(event listener)

```
// Set up our event listener to run the startup process
// once loading is complete.
startbutton.addEventListener('click', function(ev){
  startbutton.disabled = true;
  startup();
  ev.preventDefault();
}, false);
clearphoto();
```

2.5. JavaScript Hands-On

capture.js

(startup)

```
function startup() {
  navigator.mediaDevices.getUserMedia(
    {
      video: {width: {exact: 640}, height: {exact: 360}},
      audio: false
    }
  ).then(function(stream) {
    video.srcObject = stream;
    video.play();
  }).catch(function(err) {
    console.log("An error occurred: " + err);
  });

  video.addEventListener('canplay', function(ev){
    if (!streaming) {
      height = video.videoHeight / (video.videoWidth/width);

      // Firefox currently has a bug where the height can't be read from
      // the video, so we will make assumptions if this happens.

      if (isNaN(height)) {
        height = width / (4/3);
      }

      video.setAttribute('width', width);
      video.setAttribute('height', height);
      canvas.setAttribute('width', width);
      canvas.setAttribute('height', height);
      streaming = true;
    }
  }, false);

  requestAnimationFrame(takepicture);
}
```

2.5. JavaScript Hands-On

capture.js

(takepicture)

```
function takepicture() {
  var context = canvas.getContext('2d');
  if (width && height) {
    canvas.width = width;
    canvas.height = height;
    context.drawImage(video, 0, 0, width, height);

    var imgData = context.getImageData(0, 0, width, height);

    var frame = new clova.Frame(imgData.data,
                                width,
                                height,
                                clova.Format.RGBA_8888);

    var getBoundingBox = document.getElementById('bounding_box').checked;
    var getContour = document.getElementById('contour').checked;
    var getAngle = document.getElementById('angle').checked;

    var flag = 0;
    if(getBoundingBox) {
      flag |= clova.BOUNDING_BOXES;
    }
    if(getContour) {
      flag |= clova.CONTOURS;
    }
    if(getAngle) {
      flag |= clova.EULER_ANGLES;
    }
  }
}
```


2.5. JavaScript Hands-On

capture.js

(takepicture)

```
var options = new clova.FaceOptionsBuilder()
    .setInformationToObtain(flag)
    .build();
var faces = clovaSee.runForFace(frame, options).faces();

context.lineWidth = 1.5;
context.strokeStyle = 'red';
var idx = document.querySelector('input[name="selector"]:checked').value;
for (var i=0; i < faces.size(); ++i) {
    var face = faces.get(i);
    if(getBoundingBox) drawBoundingBox(context, face);
    drawMask(context, face, masks[idx]);
    if(getContour) drawContour(context, face);
    if(getAngle) drawAngle(context, face);
}

var data = canvas.toDataURL('image/png');
photo.setAttribute('src', data);

options.delete();
frame.delete();
faces.delete();

} else {
    clearphoto();
}

requestAnimationFrame(takepicture);
}
```

2.5. JavaScript Hands-On

Result

```

7 </head>
8 <body>
9   <div class="contentarea">
10     <div class="content">
11       <video id="video" hidden=true>Video stream not available.</video>
12       <button id="startbutton">Start</button>
13       <div class="features">
14         <label><input type="checkbox" id="contour" name="contour">contour</label>
15         <label><input type="checkbox" id="angle" name="angle">angle</label>
16       </div>
17       <div class="selector">
18         <label><input type="radio" id="emoji_0" name="selector" value="0"/>emoji_0</label>
19         <label><input type="radio" id="emoji_1" name="selector" value="1"/>emoji_1</label>
20         <label><input type="radio" id="emoji_2" name="selector" value="2"/>emoji_2</label>
21         <label><input type="radio" id="emoji_3" name="selector" value="3" checked="checked"/>none</label>
22       </div>
23     </div>
24     <div class="output">
25       <canvas id="canvas" hidden=true</canvas>
26       <img id="photo" alt="The screen capture will appear in this box.">
27     </div>
28   </div>
29 </body>
30 </html>

```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

zsh - javascript

python3 npx

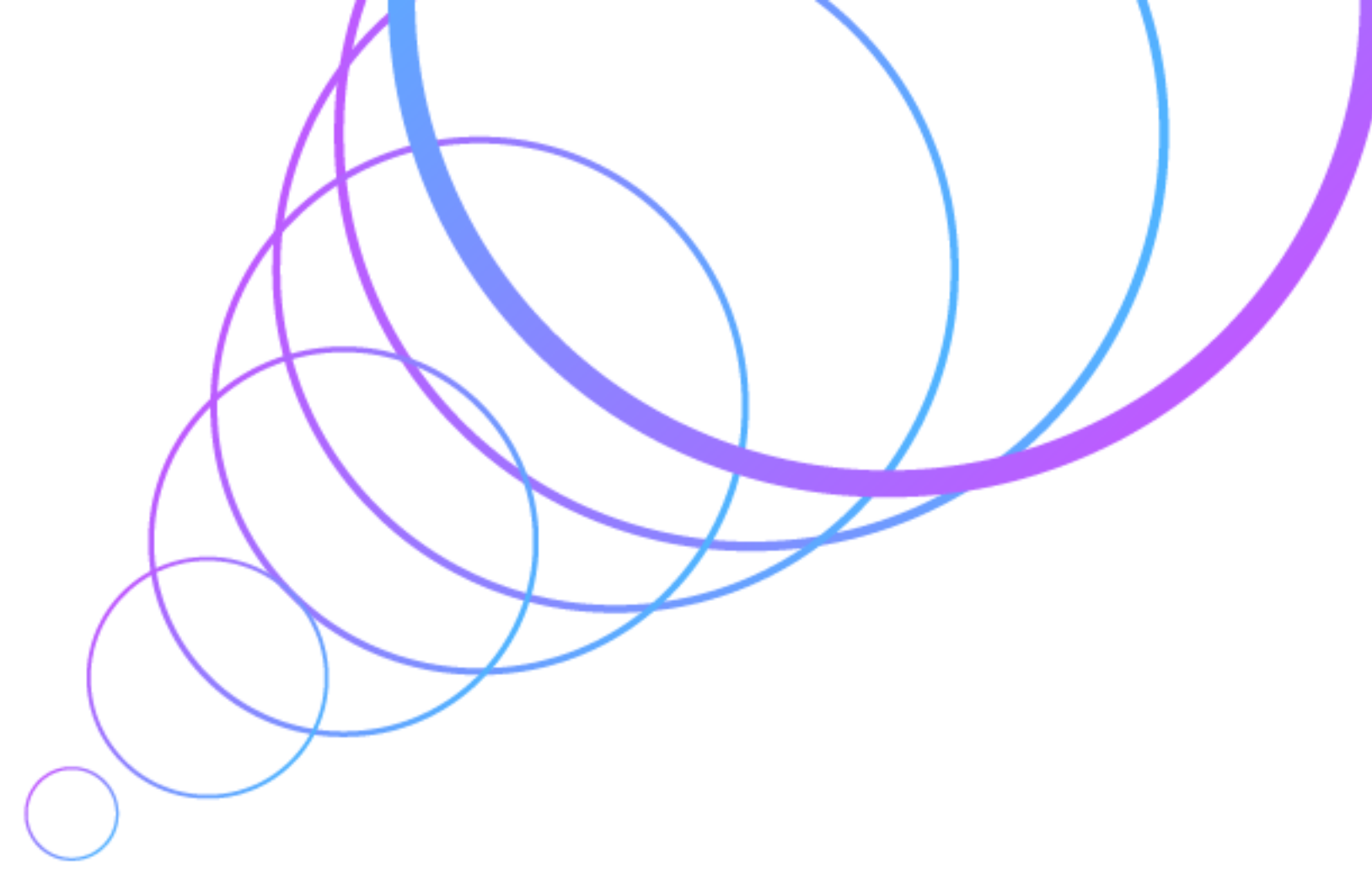
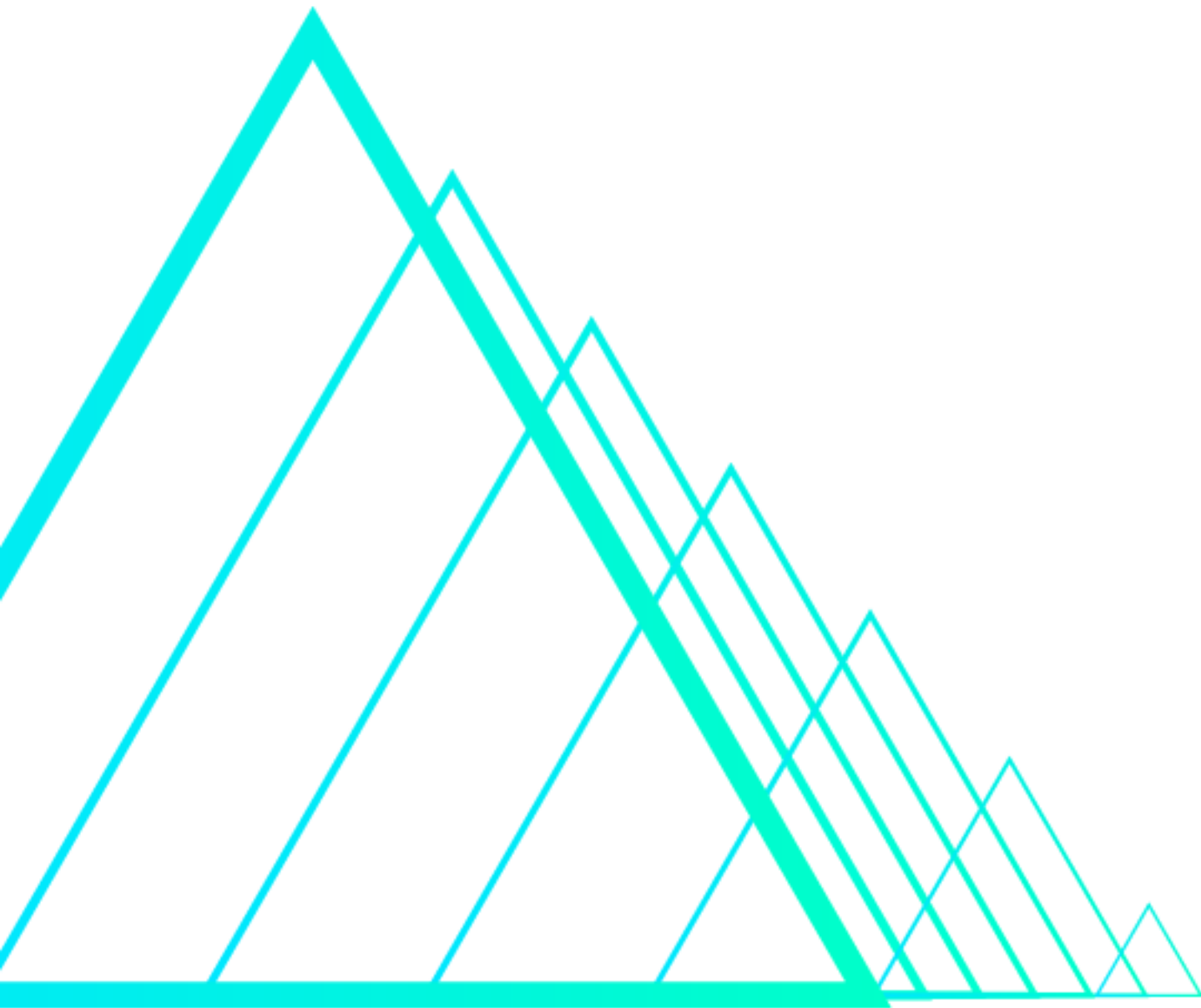
Python 3.9.6 64-bit ('devview_2021-FYBE5rkn': pipenv) Jun 6 Live Share docker-desktop default Kubernetes Ln 16, Col 13 Spaces: 2 UTF-8 LF HTML

3. Closing

3. 여기까지 오시느라 고생하셨습니다

당장 사용해보고 싶다면?

- <https://github.com/naver/clova-face-kit>



Thank You

dl_clova_partnership@navercorp.com

