

# 1. 감정상태 예측 SW

## 1.1. 감정상태 예측 SW 구조

PBR 프레임워크의 감정상태 상태 및 변화 검출/추적 모듈의 전체 구조는 그림 1과 같다. 감정상태 예측 SW는 사용자의 영상 정보를 통해 감정상태를 예측하는 SW로 얼굴 검출 블록, 표정 분석 블록, 그리고 감정상태 예측 블록으로 구성되어 있다.

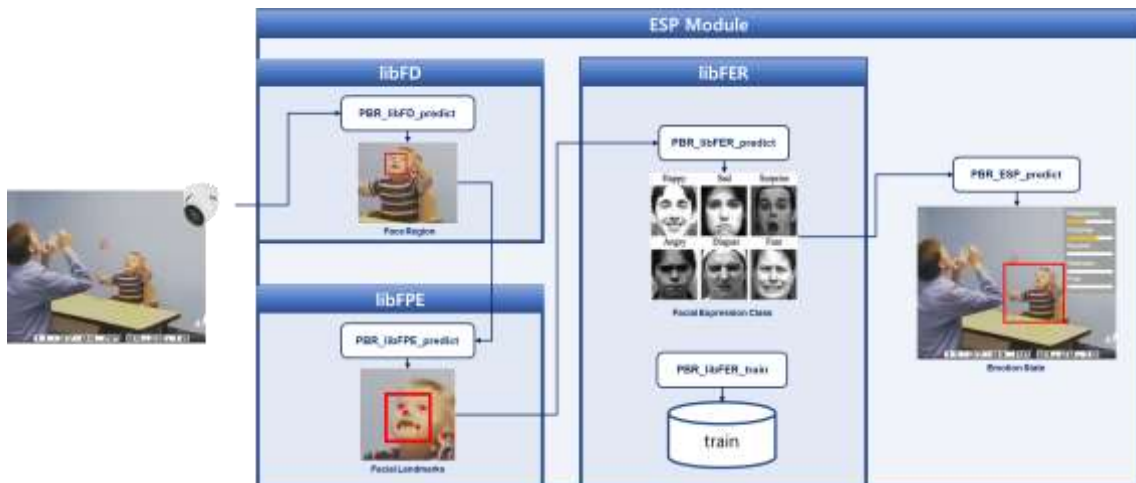


그림 1. 감정상태 예측 및 변화 검출/추적 모듈 구조도

## 1.2. 블록별 사용 라이브러리

감정상태 예측 SW는 PBR 프레임워크의 Core 모듈 중 하나인 감정상태 및 변화 검출/추적 모듈의 얼굴 정보를 이용한 감정 분류 기술이다. 감정상태 예측 SW는 얼굴 검출 및 추적(libFD) 라이브러리, 얼굴 포즈 추정(libFPE) 라이브러리, 표정 분류/인식(libFER) 라이브러리를 조합하여 개발되었다.

## 2. 기능별 API

### 2.1. 고수준 (High-level) API

#### 2.1.1. 감정상태 예측 API

<b>Function Name</b>	PBR_ESP_predict		
<b>Prerequisites</b>	PBR_libFER_train 을 이용하여 학습된 모델 파일		
<b>Prototype</b>	PBR_ESP_predict (cv_img)		
<b>Input</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	cv2::image	cv_img	표정 인식을 수행할 입력 영상
<b>Return</b>	<b>Type</b>	<b>Value</b>	<b>Meaning</b>
	int	0: Surprise 1: Fear 2: Disgust 3: Happiness 4: Sadness 5: Anger 6: Neutral	얼굴 분석을 통해 인식한 표정의 레이블
<b>Remark</b>	PBR 패키지의 ESP 모듈의 함수로 입력 영상으로부터 얼굴 검출을 수행하고, 검출된 얼굴의 표정을 분석하여, 7 가지 표정(행복, 슬픔, 놀람, 화남, 역겨움, 두려움, 중립)을 인식한다.		

\* PBR\_ESP\_predict 테스트 코드

```
import cv2
from ESP_api import PBR_ESP_predict

label_names = ['Surprise', 'Fear', 'Disgust', 'Happiness', 'Sadness', 'Anger', 'Neutral']

def main():

    """main function

    Note: main function for a facial expression recognition task

    """

    cv_img = cv2.imread('./libFER/fer_test.jpg')

    ret = PBR_ESP_predict(cv_img)
    print(label_names[int(ret)])

if __name__ == '__main__':
    main()
```

## 2.2. 저수준 (Low-level) API

### 2.2.1. 얼굴 검출 API

<b>Function Name</b>	PBR_libFD_predict		
<b>Prerequisites</b>	None		
<b>Prototype</b>	PBR_libFD_predict(cv_img)		
<b>Input</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	cv2::image	cv_img	얼굴 검출을 수행할 입력 영상
<b>Return</b>	<b>Type</b>	<b>Value</b>	<b>Meaning</b>
	List of list	[[left, top, right, bottom], ...]	얼굴의 사각영역을 나타내는 좌표 리스트, 여러 개의 사각형 영역일 경우 여러 개의 리스트를 출력
<b>Remark</b>	PBR 패키지의 libFD 라이브러리의 함수로 얼굴 검출을 수행하여 얼굴의 사각영역을 출력한다.		

### 2.2.2. 얼굴 포즈 추정 API

<b>Function Name</b>	PBR_libFPE_predict		
<b>Prerequisites</b>	None		
<b>Prototype</b>	PBR_libFD_predict(cv_img)		
<b>Input</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	cv2::image	cv_img	얼굴 포즈를 추정할 입력 영상
<b>Return</b>	<b>Type</b>	<b>Value</b>	<b>Meaning</b>
	List of list	[[x1, x2], ...]	얼굴의 랜드마크를 나타내는 좌표 리스트, 사용할 모델에 따라서 랜드마크의 개수가 정해져 있다.
<b>Remark</b>	PBR 패키지의 libFPE 라이브러리의 함수로 얼굴 포즈 추정을 수행하여 얼굴의 랜드마크 좌표를 출력한다.		

\* PBR\_libFD\_predict, PBR\_libFPE\_predict 테스트 코드

```
import cv2
from libFD.api import PBR_libFD_predict
from libFD.api import PBR_libFPE_predict
def main():
    """main function
    Note: main function for a face detection task
    """
    cv_img = cv2.imread('./libFD/fd_test.jpg')
    bbox = PBR_libFD_predict(cv_img)
    facial_landmarks = PBR_libFPE_predict(cv_img)
    cv_img = visualize(cv_img, bbox, facial_landmarks)
    cv2.imshow('test', cv_img)
    cv2.waitKey(0)

def visualize(cv_img, bbox, landmarks):
    """visualize function
    Note: visualization function using facial points and bounding boxes
    Arguments:
        cv_img (opencv image): image to visualize bounding boxes and landmarks
        bbox(list of list): [[left, top, right, bottom],...]
        landmarks(list of list): [[x, y],...]
    Returns:
        cv_img (opencv image): a visualized image
    """
    for b in bbox:
        cv2.rectangle(cv_img, (int(b[0]), int(b[1])), (int(b[2]), int(b[3])), (255, 255,
255), 1)
    for p in landmarks:
        for i in range(5):
            cv2.circle(cv_img, (int(p[i]), int(p[i + 5])), 1, (0, 255, 0), -1)
    return cv_img

if __name__ == '__main__':
    main()
```

### 2.2.3. 표정 분석 API

<b>Function Name</b>	PBR_libFER_train		
<b>Prerequisites</b>	표정 데이터 셋과 데이터 경로 및 레이블 정보 파일		
<b>Prototype</b>	PBR_libFER_train()		
<b>Input</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
argparser 기반 입력 파라미터			
<b>Return</b>	<b>Type</b>	<b>Value</b>	<b>Meaning</b>
None			
<b>Remark</b>	PBR 패키지의 libFER 라이브러리의 함수로 표정 데이터셋을 이용하여 7 가지 표정(행복, 슬픔, 놀람, 화남, 역겨움, 두려움, 중립)을 학습한다.		

\* PBR\_libFER\_train 테스트 코드

```
from libFER.api import PBR_libFER_train

def main():
    """main function
    Note: main function to train facial expression recognition
    """
    PBR_libFER_train()

if __name__ == '__main__':
    main()
```



<b>Function Name</b>	PBR_libFER_predict		
<b>Prerequisites</b>	PBR_libFER_train 을 이용하여 학습된 모델 파일		
<b>Prototype</b>	PBR_libFER_predict(cv_img)		
<b>Input</b>	<b>Type</b>	<b>Name</b>	<b>Meaning</b>
	cv2::image	cv_img	표정 인식을 수행할 입력 영상
<b>Return</b>	<b>Type</b>	<b>Value</b>	<b>Meaning</b>
	int	0: Surprise 1: Fear 2: Disgust 3: Happiness 4: Sadness 5: Anger 6: Neutral	얼굴 분석을 통해 인식한 표정의 레이블
<b>Remark</b>	PBR 패키지의 libFER 라이브러리의 함수로 7 가지 표정(행복, 슬픔, 놀람, 화남, 역겨움, 두려움, 중립)을 인식한다.		

\* PBR\_libFER\_predict 테스트 코드

```
import cv2
from libFER.api import PBR_libFER_predict
label_names = ['Surprise', 'Fear', 'Disgust', 'Happiness', 'Sadness', 'Anger', 'Neutral']

def main():
    """main function
    Note: main function for a facial expression recognition task
    """

    cv_img = cv2.imread('./libFER/fer_test.jpg')
    ret = PBR_libFER_predict(cv_img)
    print(label_names[int(ret)])

if __name__ == '__main__':
    main()
```

**Copyright © 2021 ETRI**

이 문서의 내용을 임의로 전재 및 복사할 수 없으며, 이 문서의 내용을 부분적으로라도 이용 또는 전재할 경우, 반드시 저자인 한국전자통신연구원의 서면 허락을 취득하여야 한다.