RT コンポーネント操作マニュアル

画像処理を用いた謎解き体験
コンポーネント群

[謎解き×Robot Technology]

2021 年 1 月 19 日(水)

芝浦工業大学

デザイン工学部デザイン工学科

プロジェクト演習９　４Ｑ４班

遠藤巧人(cy19078)，小田哲也(cy19225)，谷尾美亜(cy19228)

[目次]

1 本コンポーネントの概要

1.1 開発の背景

4Q のプロジェクト演習 9 では「ロボットテクノロジー×謎解き」というテーマをいただきました。今回のチームメンバーは 3 人で、UI・UX デザイナーが 1 名，エンジニア(ソフトウェア)が 2 名という構成であり「3 人のチームメンバーの能力(can)」及び「作り上げたい世界観(will)」の両方向から制作内容を考えた結果、今回の「画像処理をふんだんに用いた謎解きコンポーネント群」を開発する運びとなりました。

1.2 開発環境

本コンポーネント群は全て Windows 環境にて開発を行なっております。
開発環境は以下の通りです。

・Windows10(64bit)

・RT ミドルウェア: OpenRTM-aist-1.2.2-RELEASE(C++, Python)

・Python 実行環境: Python 3.8.10

また、開発したコンポーネントにはそれぞれ以下の依存ライブラリを使用しています。

・ゲームコンポーネント(GameRunner)
　pygame　2.1.0
・文字列入力コンポーネント(StringIn)
　無し
・画像取得コンポーネント(PhotoCapture)
　numpy　1.21.4
・正解判定コンポーネント(AnswerJudge)
　numpy　1.21.4
　pytorch　1.10.0
・二値化コンポーネント(Thresholding)
　OpenCV 4.5.4.60
　numpy　1.21.4
・ポーズ分類コンポーネント(ImageClassify)
　numpy　1.21.4
　pytorch　1.10.0

tensorflow 2.7.0
tensorflow-hub 0.12.0
matplotlib 3.5.0

・ポーズ推定コンポーネント(Movenet)
tensorflow 2.7.0
tensorflow-hub 0.12.0
numpy　1.21.4
OpenCV 4.5.4.60
matplotlib 3.5.0

・物体検知コンポーネント(SSD_predict)
numpy　1.21.4
OpenCV 4.5.4.60
pytorch　1.10.0
matplotlib 3.5.0

1.3 本稿で利用しているコンポーネント群(概要)

ゲームの進行や基本的な操作に関わるコンポーネント群としては以下のコンポーネントを開発しました。

・ゲームコンポーネント(GameRunner)
ゲーム画面表示及び他のコンポーネントへ指示を出すコンポーネント。
通常のプログラムでいう main のような役割を担う。

・文字列入力コンポーネント(StringIn)
コンソールから文字列を入力するためのコンポーネント。
1 問目の解答に使用。

・画像取得コンポーネント(PhotoCapture)
起動しているカメラから任意のタイミングで画像を取得するためのコンポーネント。
タイマー機能を搭載しており，シャッターが押されてから任意の時間差で画像を取得。

・正解判定コンポーネント(AnswerJudge)
謎解き用に他のコンポーネントが出力結果をもとに正解判定を行うコンポーネント。
この出力によってゲームの画面遷移が変わる。

また、今回の謎解きギミックの特徴である画像処理関連のコンポーネントとして以下のコンポーネントを開発しました。

・二値化コンポーネント(Thresholding)

入力された数値を元に画像の二値化を行うコンポーネント。

・ポーズ分類コンポーネント(ImageClassify)

入力された画像が用意したラベルの中でどのポーズかを推定するコンポーネント。

予測には 3 種類のモデルを用意。

・ポーズ推定コンポーネント(Posenet)

入力された画像から映っている人の姿勢を推定するコンポーネント。

・物体検知コンポーネント(SSD_predict)

入力された画像の中から定められた物体の位置と種類,個数を特定するコンポーネント。

・顔検出, 識別コンポーネント(FacialRecognition)

入力された画像から人の顔領域を検出及び特定の人物を識別するコンポーネント

2 各コンポーネントの説明

2.1 基本コンポーネント群

基本コンポーネント群は、謎解きの中心(繋げ役)となるゲームコンポーネントと、画像処理以外の基本的な操作を提供するコンポーネント群です。

2.1.1 ゲームコンポーネント(GameRunner)

GameRunner は、ゲーム画面の表示，進行，他コンポーネントへの動作の開始を合図する今回の一連の謎解き体験の中心となるコンポーネントです。

・InPort

名称	型	説明
detection_is_true	RTC::TimedBoolean	ポーズ分類問題の正解判定
ssd_is_true	RTC::TimedBoolean	SSD 問題の正解判定
binarization_is_true	RTC::TimedBoolean	二値化問題の正解判定
movenet_is_true	RTC::TimedBoolean	ポーズ問題の正解判定

・OutPort

名称	型	説明
button	RTC::TimedLong	カメラのシャッターの役割 具体的には”9”が入る

・Configuration 変数

名称	型	説明
なし	なし	なし

2.1.2 文字列入力コンポーネント(StringIn)

StrinIn は string 型のデータをコンソール上で入力し、それを出力するコンポーネントです。

・InPort 変数

名称	型	説明
なし	なし	なし

・OutPort

名称	型	説明
out	RTC::TimedString	コンソール上で入力された文字列を出力するポート

・Configuration 変数

名称	型	説明
なし	なし	なし

2.1.3 画像取得コンポーネント(PhotoCapture)

PhotoCapture は Camera のフレームを常時受け取り、特定のキー(“9”に設定)が入力された際に画像として取得し別のコンポーネントへと出力するコンポーネントです。

・InPort

名称	型	説明
in_image	RTC::CameraImage	カメラから送られたフレーム (例: 入力元「OpenCVCamera」「MFCamera」等)
button	RTC::TimedLong	画像を取得する合図 この入力が数字の”9”であった場合に画像を取得する それ以外の入力に対してはスルー

・OutPort

名称	型	説明
out_image	RTC::CameraImage	取得した画像

・Configuration 変数

名称	型	説明
delay	int	画像取得の合図が送られてから画像を実際に取得するまでの時間差。一般的なカメラの「タイマー」に相当する。

2.1.4 正解判定コンポーネント(AnswerJudgment)

AnswerJudgment は謎解き用の各問題に対する解答を受け取り，それらの正解判定を行うコンポーネントです。

・InPort

名称	型	説明
predict_label	RTC::TimedLong	画像分類器が出力した予測値(label)。 ex) 0, 1, …, 10
predict_detections	RTC::TimedFloatSeq	SSD を行うコンポーネントが出力した予測値を一次元にした配列。配列の元形状は(1, 21, 200, 5)であるため，この入力の形状は(21*200*5,)=(21000,)である。一次元にする前の予測値の詳細に関しては SSD_predict を参照。
movenet_score	RTC::TimedFloat	Movenet が出力した正解画像と入力画像の類似度。
binarization_str	RTC::TimedString	文字列。問題 1 の答えを入力する際に使用。

・OutPort

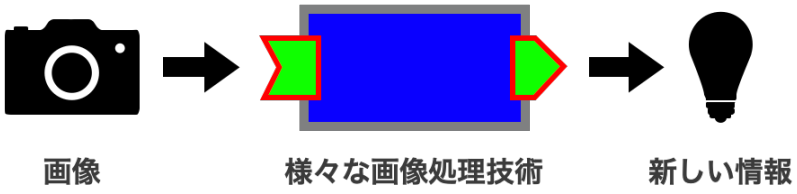
名称	型	説明
detection_judge	RTC::TimedBoolean	ポーズ分類問題の正解判定
SSD_judge	RTC::TimedBoolean	SSD 問題の正解判定
movenet_judge	RTC::TimedBoolean	ポーズ問題の正解判定
binarization_judge	RTC::TimedBoolean	二値化問題の正解判定

・Configuration 変数

名称	型	説明
なし	なし	なし

2.2　画像処理コンポーネント群

画像処理コンポーネント群は今回の私たちの制作の中でキモになるコンポーネントの集まりです。これらのコンポーネント群は全て画像を入力として受け取り、各々の画像処理を実施，得られた情報をアウトプットします。



2.2.1 二値化コンポーネント(Thresholding)

Thresholding は入力画像をキーボードから入力された閾値で処理し、処理後の画像を出力する。閾値より大きな値はそのまま、それ以外の値は 0 に置き換えます。

・InPort

名称	型	説明
input_img	RTC::CameraImage	閾値処理を行う画像を入力する。
threshold	RTC::TimedLong	RTC::TimedLong 型の閾値を入力する。

・OutPort

名称	型	説明
output_img	RTC::CameraImage	キーボードから入力された閾値より大きい値をそのまま、それ以外を 0 に置き換えた画像を出力する。

・Configuration 変数

名称	型	説明
なし	なし	なし

2.2.2 ポーズ分類コンポーネント(ImageClassify)

ImageClassify は予測対象の画像を受け取り，その画像が用意した正解の内，どのポーズに当てはまるかを推論するコンポーネントです。予測に使うモデルは 3 種類用意しており，vgg16 の転移学習による推論, EfficientNetV2 の転移学習による推論, Movenet を用いた最も近いポーズの予測を使用することができます。

また、参加者にポーズをとってもらう問題に関しては次の Movenet コンポーネントを使用することも考えられます。Movenet では特定の pose にかなり近い pose を取らない限り正解にはなりませんが、Image Classify では取った pose が他ラベルに比べて正解のラベルに近ければ正解と判定されるため、難易度としてはこちらの方が簡単になります。

・InPort

名称	型	説明
input_image	RTC::CameraImage	予測(分類)対象の画像

・OutPort

名称	型	説明
label	RTC::TimedLong	モデルが予測したポーズのラベル

・Configuration 変数

名称	型	説明
なし	なし	なし

参考: Tensorflow Hub EfficientNetV2 公式ドキュメント及び google colab notebook

2.2.3 ポーズ推定コンポーネント(Movenet)

Movenet は入力画像に映る人物の pose(17 箇所の座標及び信頼度)を推定した後，用意した画像との pose の類似度を計算，出力するコンポーネントです。

・InPort

名称	型	説明
input_img	RTC::CameraImage	ポーズ推定対象の画像

・OutPort

名称	型	説明
output_score	RTC::TimedFloat	正解画像と入力画像の pose 類似度。 類似度は input_img に対して Movenet が予測した身体の各ポイントの座標と，正解画像に対して Movenet が予測した身体の各ポイントの座標をそれぞれ min-max scaling したのち ユークリッド距離を計算。

・Configuration 変数

名称	型	説明

なし	なし	なし

参考: Tensorflow Hub Movenet 公式ドキュメント及び google colab notebook

2.2.4 物体検知コンポーネント(SSD_predict)

SSD_predit は入力された画像に対して物体検知を行い，一定の信頼度以上の物体の label 及び BBox の座標値とその信頼度を出力するコンポーネントです。

・InPort

名称	型	説明
input_img	RTC::CameraImage	SSD による予測の対象になる画像

・OutPort

名称	型	説明
detections	RTC::TimedFloatSeq	SSD を行うコンポーネントが出力した予測値を一次元にした配列。SSD の出力は 予測対象のクラス 20class + 背景 1class = 21 1 クラスにおける BBox の最大数 = 200 BBox の四隅の座標(4 点)+信頼度(1score) = 5 であるため，この入力の形状は(21*200*5,)=(21000,) となる。

・Configuration 変数

名称	型	説明
なし	なし	なし

参考: 物体検出と GAN、オートエンコーダー、画像処理入門(チーム・カルポ)

2.2.5 顔検出，識別コンポーネント(FacialRecognition)

FacialRecognition はカメラから入力された画像の顔検出と特定の顔の識別をします。
顔が特定された人は、BBox が赤くなり、温度表示が熱に設定されます。
それ以外の人は、BBox が青くなり、体温表示が常温に設定されます。

・InPort

名称	型	説明
in	RTC::CameraImage	RTC::CameraImage 型の画像を入力する。
bool	RTC::TimedBoolean	True が入力されると、映像の出力を開始する。

・OutPort

名称	型	説明
output_img	RTC::CameraImage	カメラから受信した画像に BBox と体温表示を付加して出力する。

・Configuration 変数

名称	型	説明
なし	なし	なし

3 コンポーネント群の使用方法

この節では GameRunner を中心としたコンポーネント群全体の使用方法を説明します。GameRunner 中には全 4 問問題が用意されていますので、それに沿ってコンポーネント群の使用方法を説明します。とはいえ、今回制作したコンポーネントのほとんどには Configuration 変数が無く使用の際にセッティングする場所はほとんどありません。そのため各問題で使用するコンポーネントとその繋ぎ方に気をつければ簡単にシステムを動かすことができます。

3.1 起動から問題 1 終了までのコンポーネントの使い方

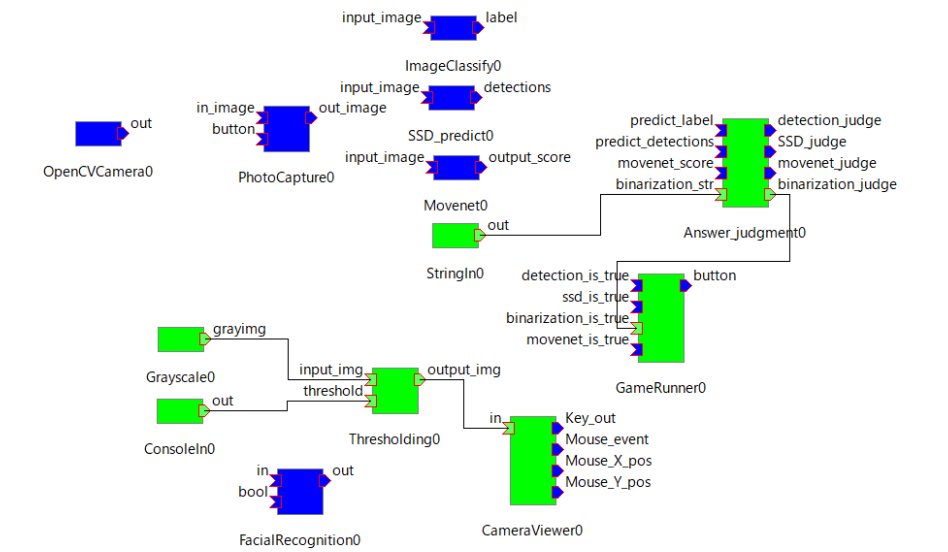
[使用するコンポーネント]

1. Grayscale
2. Thresholding
3. StringIn
4. GameRunner
5. Answer_judgment
6. GameRunner

[sample component]

7. ConsoleIn
8. CameraViewer

[コンポーネントの繋ぎ方]



3.2 問題 2 終了までのコンポーネントの使い方

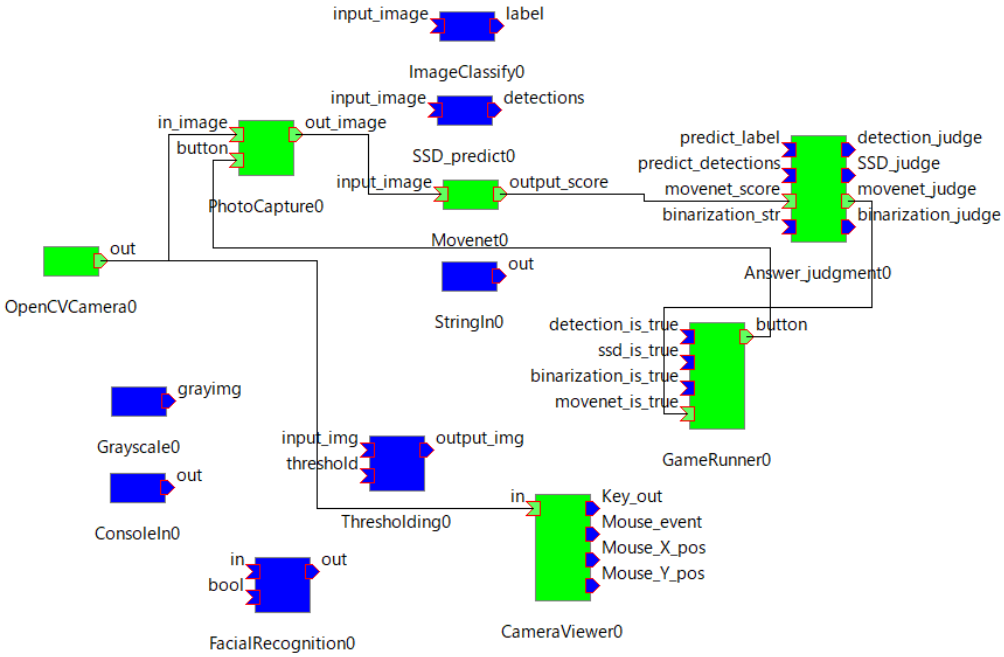
[使用するコンポーネント]

1. PhotoCapture
2. Movenet(もしくは ImageClassify)
3. Answer_judgment
4. GameRunner

[sample component]

5. OpenCVCamera
6. CameraViewer

[コンポーネントの繋ぎ方]



3.3 問題 3 終了までのコンポーネントの使い方

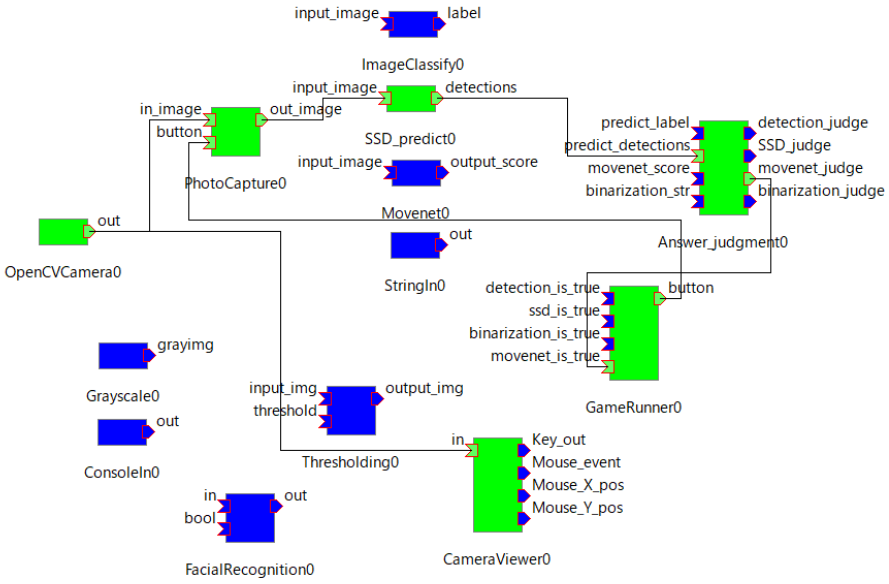
[使用するコンポーネント]

1. PhotoCapture
2. SSD_predict
3. Answer_judgment
4. GameRunner

[sample component]

5. OpenCVCamera
6. CameraViewer

[コンポーネントの繋ぎ方]



3.4 問題 4 及びエンディングまでのコンポーネントの使い方

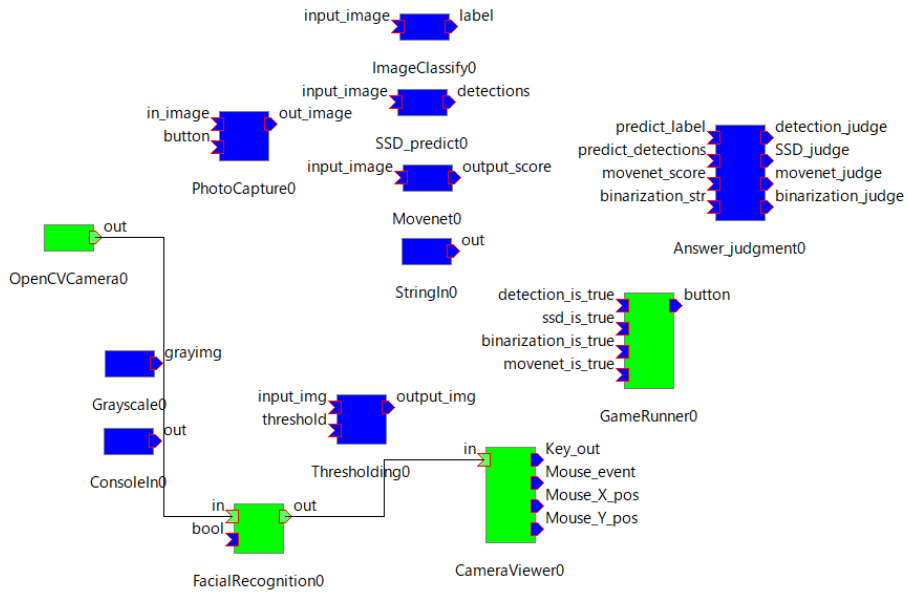
[使用するコンポーネント]

1. PhotoCapture
2. SSD_predict
3. Answer_judgment
4. GameRunner

[sample component]

5. OpenCVCamera
6. CameraViewer

[コンポーネントの繋ぎ方]



4 改善の余地について

4.1 改善の余地について

さて、これまでの節では今回私たちが作ったコンポーネント群についての説明, 使い方を書いてきましたが、最後に多分にある改善点の中から特に大きな 2 点について、記させて頂きたいと思います。

1. Configuration 変数の使用
   節 3 の冒頭で述べた通り今回作成した私達のコンポーネントにはほとんど Configuration 変数が存在しません。これは Configuration 変数にする変数がプログラム上になかった訳では無く、それらの変数をプログラム上にそのまま埋め込んでしまっているからです。ImageClassify に使用するモデルの種類や Answer_judgment 内の正解値, GameRunner 中の BGM・効果音音量, SSD_predict の判定閾値等、Configuration 変数にすべき変数を上げるとキリがありません。プログラムを書いている身からするとプログラムを直接変えたほうが楽であるために生まれてしまった落ち度だと考えています。

2. PhotoCapture の送り先指定
   今回作成した一連の謎解き体験において、問題と問題の間ではコンポーネントの組み替えが必要になります。特に PhotoCapture と各画像処理コンポーネントは全てを繋いでおくと自分の番で無くても推論を実施してしまうため常につなげておくということができません。実際に使用する場面を考えるといちいち問題のたびにコンポーネントの配線を変える時間や手間はなるべく省きたいところです。改善の仕方としては GameRunner の現在位置(ページ数)を受け取り、適宜送り先を変えるコンポーネントを PhotoCapture と各モデルとの間に挟むなどが考えられると思います。

3. game 画面における謎のフリーズ
   game 画面の中で 1 画面だけ、時折フリーズを起こしてしまう画面が存在しています。原因は今の所不明です。