

カイクレダイコンの育成を管理する シミュレーションの製作

<資料の読み順について>

1. 資料1(本書):「カイクレダイコンの育成を管理するシミュレーションの製作」
2. 資料2:「事前知識」
3. 資料3:「カイクレダイコンの育成に必要な情報得る画像処理」
4. 資料4:「シミュレーションソフトの主な設計について」
5. 資料5:「シミュレーションソフトの操作方法」

<必要な OS とソフトウェアについて>

OS : window7・windows8・windows8.1・windows10 推奨

※x64 対応

コードレビューの際に必要なソフトウェア:

- ・「Visual Studio 2015 community」 C#,C++,C++/CLI
- ・C ファイルが開けられるエディタ

<目的>

IoT を想定。カメラからカイワレダイコンの育成を監視し、育成状況を利用者に提供するシステムを構築する。しかし、費用や時間が限られ、また一人で行うため、費用・工数を考慮した開発とする。よって、現段階ではカイワレダイコンの育成を管理する装置と装置から送られてきたデータを処理・蓄積するサーバ・利用者に提供するインターフェースをシミュレータとし、一連の流れをシミュレーションするソフトを開発。

<動機>

私は中学3年生の頃、低血糖症による睡眠障害にかかり治療方法が確立されていなかったため、食生活に重視し、健康食品を多く食べるようになった。その一つ、カイワレダイコンを食べていた。その時にカイワレダイコンについて色々な事を知った。カイワレダイコンの特性は栄養価が高く育てるのが簡単な植物である事だ。また、中学3年生時に作成した理科自由研究「光と植物の関係」(資料3)もカイワレダイコンを対象にして実験をし、結論である「太陽光の力を使わずとも育てられる植物育成 LED がある」という事を踏まえて何か作れないかと考えた。そして、カイワレダイコンの知識・植物育成 LED・コンピュータ技術を生かし、「どこでも育てられ、どこにいても育成状況が分かるシステム」を構築したいと考えた。また、私が持っているプログラミング技術を最大限にアピールができる方法だ。中学3年生時に作成した理科自由研究「光と植物の関係」は第18回京都サイエンスコンテストで佳作を受賞した。

<対象>

作成するプログラムについてはシミュレーションですので、対象者は開発者向け（自身など）である。しかし、完成した時の対象は「スマートフォンなどから育成状況を知りたい人」、「自分で育てる時間がない人」である。

＜作成したプログラムの一覧＞

実行方法・操作手順については詳細に記載してある

1. 実験 1.exe

目的：植物を測る上で 2 cm、3 cm… 13 cm の境界線が画像のどの位置なのかをプログラムによりピクセル単位で求める。

2. 植物画像処理.exe (OpenCV3.1 を使用)

目的：植物を対象に撮影した画像から植物の育成に関する情報を求める

- ①植物画像からエッジ検出をし、最長点を求める
- ②元の画像を 1/4 リサイズし、罫線のある画像を出力
- ③植物画像から求めた最長点の値から伸長を求める
- ④カイワレダイコンの各期間を求める
- ⑤植物育成 LED の入り切り求める

以上、5 つのプログラムをまとめて実行できる「植物画像処理.exe」

3. 「植物専用ライブラリ」(OpenCV3.1 を使用)

目的：「シミュレーションソフト C#」で動かすために C++/CLI、C# でライブラリ

- ・ plant_pixel.dll
- ・ plant_gridpicture.dll
- ・ plant_resizepicture.dll
- ・ plant_lenght.dll

4. 「ネットワークライブラリ」

目的：「シミュレーションソフト」でネットワークを利用するためである

以下、送信用と受信用の 2 つのライブラリを作成

- ・ SendLib.dll
- ・ ReceiveLib.dll

5. 「シミュレーション (本ソフト)」

目的：機器・サーバ・クライアントの各 3 つのプログラムを作成。

- ・ plant_machine.exe
- ・ plant_server.exe
- ・ plant_client.exe

＜開発期間＞

4 月から 8 月

＜著作権が関わるライブラリ・ファイルの使用について＞

OpenCV3.1 :

「./license/OpenCV3.1/」にライセンスについて記している。

Visual Studio :

ランタイムライブラリ(redest)を使用している

- msvcp140.dll
- conrct140.dll
- vcruntime140.dll

＜テスト用画像について＞

種子からカイワレダイコンの成長を育成していたら時間がかかるので、市販のカイワレダイコンを使用してテスト用画像を作成していく。

テスト用画像があるディレクトリ : 「./Palnt-IoT/テスト用画像/」

ファイル名について :

発芽期・測定不可は実際の伸長を記していないが、そのほか、画像は

「コンピュータで求めた伸長 (実際の伸長).JPG」をファイル名にしている。

単位は cm[小数第一位]である。

また、コンピュータで求めた伸長が同じだった場合、各番号を付与している。

※ 「./提出/bin/シミュレーション/植物育成管理機器/植物画像(テスト)/」内にあるテスト画像は参照するだけで変更はしないでください。

<コードレビューについて>

エラー非表示・ビルド等をする際の設定

①OpenCV3.1 を利用しているソースファイル.sln(VS2015) の設定：

OpenCV3.1 をダウンロードする。

まず、プログラムをビルドする際に必要なインクルードファイル・ライブラリの在所を設定

1. プロジェクトのプロパティを表示させる

2. 構成を「すべての構成」に変更。

3. インクルードディレクトリ位置の設定

「構成プロパティ」->「C/C++」->「全般」->「追加のインクルードディレクトリ」で
「./OpenCV3.1/build/include/」を設定。

※OpenCV3.1 のファルダを移動した際は、「OpenCV3.1/build/include/」のインクルードファルダのディレクトリを設定

4. ライブラリディレクトリ位置の設定

「構成プロパティ」->「リンカー」->「全般」->「追加のライブラリディレクトリ」で
「./提出/lib/OpenCV3.1/build/x64/vc14/lib」を設定。

※OpenCV3.1 のファルダを移動した際は、「OpenCV3.1/build/x64/vc14/lib」のライブラリディレクトリのディレクトリを設定

②シミュレーションソフトの各ソースファイル.sln(VS2015) の設定：

自作ライブラリを使用しているため、「参照」を更新しなければならない

1. 「ソリューションエクスプローラー」を表示させる。

2. 「ソリューションエクスプローラー」->「plant_(client・server・machine)」->
「参照」->「(エラー表示がしている各ライブラリ)」をすべて削除。

※エラー表示がしている各ライブラリを覚えておく。

もしくは、定義場所を参照

3. メニューバーの「プロジェクト」->「参照追加」をクリック・表示。

4. 「参照」の「参照(B)...」をクリック・表示。

5. 「./MyLibrary /... /」内にあるライブラリを定義場所に記述している
自作ライブラリを追加する

＜評価してもらいたい点＞

- ・「C、C++、C++/CLI、C#」と多くの言語を使用したこと
- ・一つのソフトに C++/CLI・C#(.NET 対応)の二つを利用したこと。
- ・「自作ネットワークライブラリ」は他のソフトを開発するとき汎用性があること
- ・xml を利用してデータベースとして、クライアント用ソフトに表示したこと
- ・OpenCV3.1 の一部の機能(エッジ検出など)を利用したこと
- ・カイワレダイコンを主体にした理由がしっかりあること。
- ・出来る限り工数を少なくするために、マシン・サーバ・クライアントの形態を工夫したこと
- ・カイワレダイコンの伸長を求めるプログラムは少しの誤差で実際の長さに近い値が出たこと
- ・実際にラズベリーパイを使用してこの作品を使用できる状態にしたかったが、費用の関係でシミュレーションにしたこと

など