

G-9

継続的な開発のための拡張現実型体感学習ツールの C# 化 tangible leaning with Augmented Reality technology on C#

岡本 健一郎^{*}, 山口 健[†], 吉川 浩[†]
Okamoto Kenichiro^{*}, Takeshi Yamaguchi[†], Hiroshi Yoshikawa[†]

Abstract: Augmented Reality technology can impose us sensation like a modifying thing to project CG for camera image. This software was developed with C++ at preceding study. After that, It was rewritten with Unity for the purpose of enhancing Scalability. But performance became slow. In this research, I enhance this software and cover a defect of Unity version by rewriting of C# and using new Library like OpenGL and OpenCV.

1. まえがき

拡張現実感技術はカメラで写した画像に CG を投影することでものを動かしているような感覚を得ることができる。当研究室では、この技術を利用して光学素子の操作を学習できるシステム (以降 AR ホロトレーナと呼ぶ) が先行研究にて C++ で開発されている [1]。その後、拡張性を高めるために同システムを Unity を用いて書き換えられたが、動作が重くなってしまった [2]。そこで本研究では、Visual C# に書き換えることで Unity 版の欠点を補う。同時に、OpenGL や OpenCV などのライブラリを用いることで拡張性を高め、同システムの継続的な開発を促進できるようにする。

2. 原理・方法

2. 1. ソフトウェア全体の流れについて

ソフトウェア起動時の流れを Fig. 1 に示す。レーザー光発射時に、プログラム撮影の判定に必要な値である光強度、偏光の方位角、凸レンズによりレーザー光の太さを制御する時に必要なレーザー光の半径などのパラメータを生成する。

2. 2. 各ライブラリの連動

本研究で開発する AR ホロトレーナは Visual C# という言語を用いて開発する。この言語は、GUI によってボタンやダイアログなどの一般的な WindowsForm のコントローラを配置することができ、C++ と違ってメモリの管理をコード上でしなくてよいので読みやすいという特徴がある。また本研究では OpenGL, OpenCV, NyARToolKit という 3 つのライブラリを用いたが、それぞれの関係と役割は Fig. 2 のようになっている。OpenGL には GLControl と呼ばれる互換ライブラリがあり、WindowsForm に GUI で OpenGL の描画ウィンドウを自由な位置に埋め込むことができる。OpenCV はカメラ画像の取得を容易にすることができ、NyARToolKit は Visual C# に対応した AR マーカーを用いた AR ソフトウェア開発用ライブラリである。

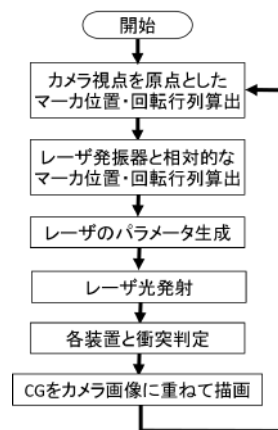


Figure 1 Software Flowchart

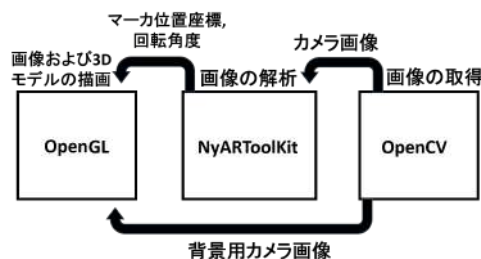


Figure 2 Relationship of Library

2. 3. 光学素子の座標の算出

カメラに映ったマーカの位置は NyARToolKit にカメラ画像データを入れることで、カメラ視点を原点とした時のマーカの変換行列を算出してくれる。素子を描画する際に各マーカ ID の変換行列を OpenGL のモデルビュー行列に掛けることでマーカ上に素子が常に描画される状態が実現する。

2. 4. レーザ発振器を原点とした座標の算出

本研究では、レーザー発振器からの光学素子の位置によってレーザー光の当たり判定をするため、カメラ視点を原点とした変換行列をレーザー発振器を原点とした変換行列に書き換える。T(m1) をレーザー発振器の変換行列、T(m2) を光学素子の変換行列とすると、レーザー発振器を原点とした他の装置の変換行列 T(m3) は Eq. (1) で表

される。

$$T(m3) = T(m1)^{-1}T(m2) \quad (1)$$

2. 5. 光学素子による光の操作

光路を操作するために、光を分岐させる PBS、光を反射させるミラーを実装する。また、光の太さを操作するために対物レンズと凸レンズを実装する。対物レンズで光の太さを広げ、凸レンズでそれをホログラム撮影に必要な平行光にする。偏光の角度を操作するために 1/2 波長板を実装する。

2. 6. 光学系構築の学習

デニシユークホログラムの光学系を Fig. 4 に示す。光学素子はレーザー発振器、対物レンズ、撮影対象、乾板を用いる。

3. 結果

3. 1. 光学素子の座標の算出

NyARToolKit を用いて OpenCV から得たカメラ画像を解析してマーカーの変換行列を算出した。変換行列を用いて WindowsForm 上に光学素子を表示している様子を Fig. ?? に示す。マーカー上に CG が乗るような形で GLControl に表示された。

3. 2. レーザ発振器を原点とした座標の算出

レーザー発振器から発射したレーザー光が対物レンズに衝突したときの様子を Fig. 5 に示す。このように、レーザー光はレーザー発振器を原点として Y 軸方向に発射されるため、対物レンズが Y 軸上にある時に衝突判定がされるようにすることで当たり判定が行えた。

3. 3. 光学素子による光の操作

PBS の実装結果を Fig. 6, ミラーの実装結果を Fig. 7 に示す。どちらも原理通り光路を変えることができているとわかる。Fig. 8 に凸レンズの実装結果を示す。対物レンズで広げた光を平行光にすることができている。波長板の実装結果を Fig. 9, Fig. 10, Fig. 11 に示す。波長板を回すことで偏光の角度が変わり、PBS から出力される 2 本の光の光強度の比率が変わっているのが分かる。

4. むすび

プログラムの作成開始時から拡張性が高くなるように意識していた。具体的には、マーカー ID とマーカーデータをフォームクラスという一つのプログラムファイルに変数として登録して、機能を書き込めば簡単に装置を追加できるため、十分な拡張性があるといえる。一方で、カメラ画像を取り込む段階でシステムのメモリ不足がしばしば発生することがあり、これは、カメラ画像の形式をシステムに取り込む際に変換するときに誤動作が起きていることが原因であると考えられる。それに加えて今回実装した装置以外の装置の追加を進めていく。

参考文献

[1] 柴田 将志:“拡張現実感技術を用いた体感学習ツールの開発”, 日本大学大学院理工学研究科修士論文 (2011).

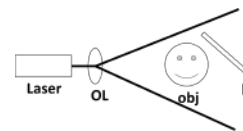


Figure 3 Denisyuk type hologram

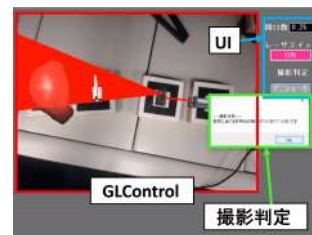


Figure 4 Software formWindow



Figure 5 OL result

Figure 6 PBS result



Figure 7 Mirror result

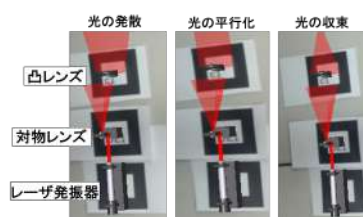


Figure 8 colimatelens result

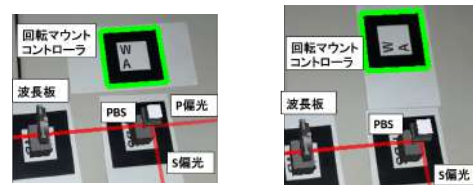


Figure 9 waveplate0°

Figure 10 waveplate22.5°

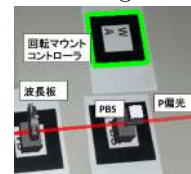


Figure 11 waveplate-22.5°

[2] 川島 吉博:“AR を用いたホログラム撮影学習ツールの開発”, 平成 27 年度日本大学理工学部卒業論文 (2015).