

iPad を用いた画像処理アプリケーションの開発

システム科学技術学部 機械知能システム学科

1年 五島 啓太

1年 佐藤 克紀

指導教員 システム科学技術学部 機械知能システム学科

助教 間所 洋和

准教授 佐藤 和人

指導補助 学部4年 西宮 享介

学部4年 水上 喬嗣

1.はじめに

iPhone アプリケーションとは、2007年にマーケットに登場した Apple 社製のタッチパネル式携帯電話 iPhone 上で動作するアプリケーションである。iPhone アプリケーションは 2010年に同社によって発売された最新式のタッチパネル式タブレット型コンピュータ iPad 上でも動作可能である。iPhone アプリケーションの特徴としては、Apple 社製のパソコンと iPhone, iPad, iPod touch などの実機と、ソフトウェア開発環境の iPhone SDK (Software Development Kit) があれば、自分で作ったアプリケーションを簡単に動作させ、インターネットを通じて広く公開できることである。

iPad は iPhone より大きな画面が特徴であり、一度に扱える情報量も多い。最近では、顔画像から戦闘力を算出するといった、ユーザを楽しませるアプリケーションが数多く開発されている。

本研究では、ユーザが楽しく感じることでできるアプリケーションを作成することを目的として、カメラを使ったアプリケーションを開発した。

2.研究内容

iPhone アプリケーションを開発するには、統合開発環境の Xcode を使用する。Xcode は、Object-C 言語でプログラミングを行うが、iPhone アプリケーション開発には専用の GUI (Graphical User Interface) を利用することができるため、プログラミングの知識が少なくてもアプリケーションを作成することができる。

アプリケーションを作成するにあたり、実機への実装が必要不可欠である。そのため、Xcode に iPad を認証させる必要があった。しかし、Xcode のバージョンが古く認証ができないため、Xcode ならびに Mac OS のアップデートを行った。図 1 に、Xcode のインターフェースを示す。

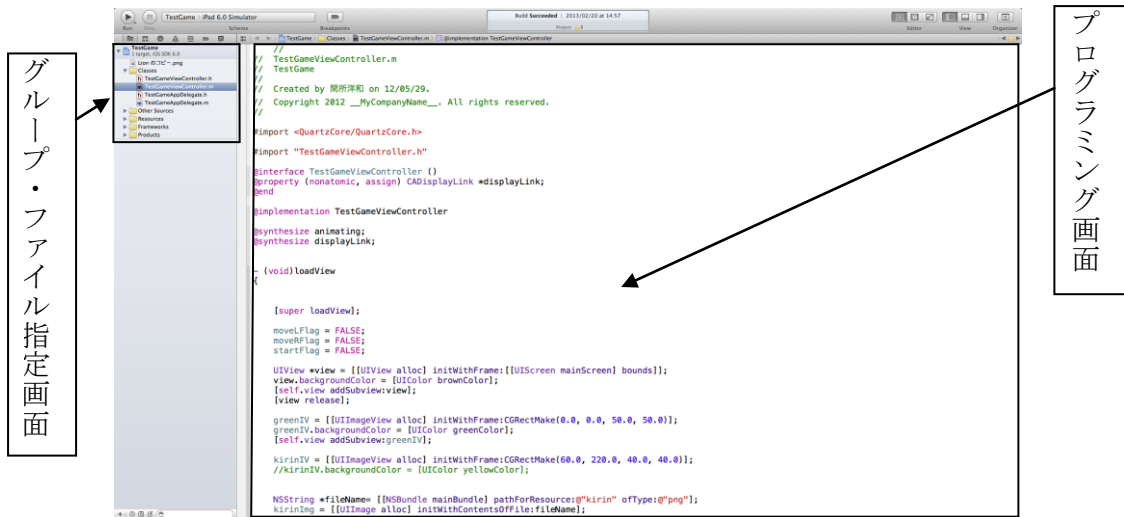
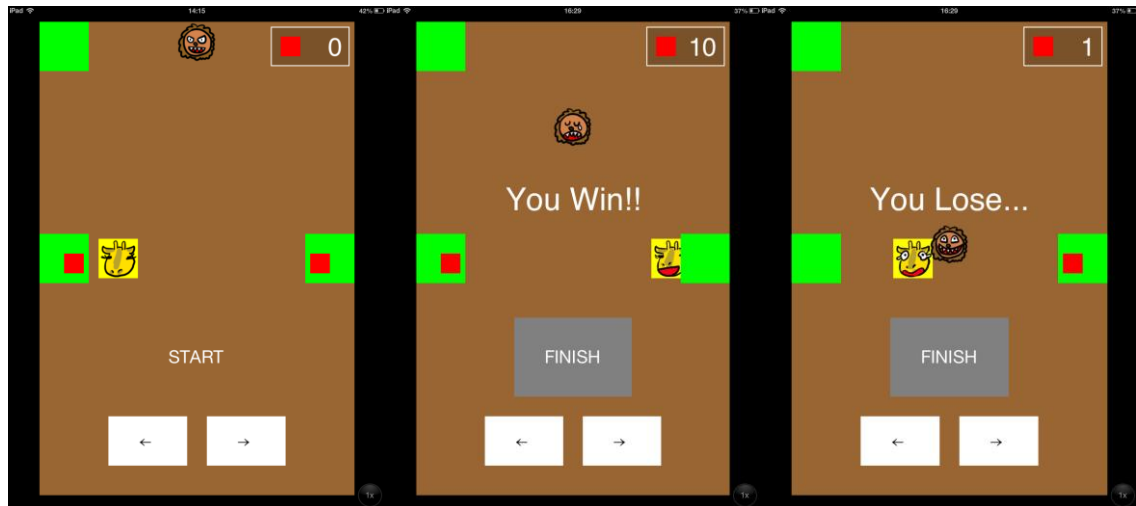


図 1. Xcode のインターフェース

3.研究成果

3.1 ゲームアプリケーションの開発

本研究を進めるにあたり、Xcode の使用方法およびプログラミングに慣れるために、簡単なゲームアプリケーションの作成を行った。作成したアプリケーションは、プレイヤーが麒麟を操作するもので、邪魔をしてくるライオンを避けつつ、左右に出現するりんごを 10 個取れば勝ち、ライオンに当たった場合は負けとなるゲームである。図 2 にゲームの画面を示す。



(a) スタート画面

(b) 勝利時の画面

(c) 敗北時の画面

図 2. ゲームの起動画面

3.2 カメラアプリケーションの開発

本研究では、ユーザが楽しめるアプリケーションとして、カメラを使ったアプリケーションを作成した。本研究で作成したアプリケーション“DoodleCamera”は、カメラで撮影中の画面を、指をペンに見立ててイラストしつつ、写真の撮影が行えるアプリケー

ションである。ペンは指定した色と線の太さに変更して描画が行われる。カメラ撮影ボタンをタップすると、落書きとカメラで撮影されている画像を合成し、iPhone,iPad のフォトアルバムに画像が保存される。図 3 に DoodleCamera の起動画面を示す。



図 3. DoodleCamera の起動画面

アプリケーション作成の流れとしてプロジェクトの作成，オブジェクトの配置，プログラムの入力の順で構成される。

このアプリケーションは、落書き画面、ペンの色選択画面、ペンの線の太さ選択画面、撮影画面の 4 画面で構成されておりそのため、オブジェクトの配置として画面の下部に撮影ボタン・色選択ボタン・クリアボタン・線の太さ選択ボタンを配置し、それぞれのボタンに対応するプログラムの作成を行った。

このアプリケーションを作成するに当たり、色の種類の少なさと、カメラの向きに着目し改良した。図 4 の(a)に追加した色、図 4 の(b)にプログラム変更前、(c)に変更後のカメラをそれぞれ示す。

また、図 5 には色の種類を追加するために書き足したプログラムを示し、図 6 にはフロントカメラからバックカメラに切り替えるために変更したプログラムを示す。



図 4. プログラムによる変更結果

```

@implementation PenColorDataSource
// 初期化メソッド。選択結果を反映するDrawViewを指定してインスタンスの初期化を行う
-(id)initWithDrawView:(DrawView*)aDrawView {
    if (self = [super init]) {
        // 引数のDrawViewインスタンスをメンバー変数に格納
        drawView = [aDrawView retain];
        // 選択肢を格納する配列を作成
        NSMutableArray *viewArray = [[NSMutableArray alloc] init];

        NSArray *colors = [NSArray arrayWithObjects:[UIColor redColor],
            [UIColor blueColor],
            [UIColor yellowColor],
            [UIColor greenColor],
            [UIColor brownColor],
            [UIColor whiteColor],
            [UIColor blackColor],
            [UIColor orangeColor],
            [UIColor cyanColor],
            [UIColor magentaColor],
            [UIColor purpleColor],
            [UIColor grayColor],
            [UIColor lightGrayColor],
            [UIColor darkGrayColor],nil];
        for (UIColor *color in colors) {

```

プログラム追加部分

図 5. 色の種類の追加

```

// 正面に配置されているカメラを取得
AVCaptureDevice *camera = [self cameraWithPosition:AVCaptureDevicePositionFront];
// カメラからの入力を作成
NSError *error = nil;
AVCaptureDeviceInput *videoInput = [[[AVCaptureDeviceInput alloc] initWithDevice:camera error:&error] autorelease];

// 画像への出力を作成
stillImageOutput = [[AVCaptureStillImageOutput alloc] init];

// 入力と出力からキャプチャーセッションを作成
AVCaptureSession *session = [[AVCaptureSession alloc] init];
[session addInput:videoInput];
[session addOutput:stillImageOutput];

```

Frontをbackに変更

図 6. カメラの向き変更

4 まとめ

本研究では iPhone,iPad のアプリケーションを作成し実機での動作確認を行った。開発したアプリケーションを研究室の先輩方に使用して頂いたところ、楽しんでもらうことができ、ユーザーを楽しませるという点は達成することができた。また、本開発を通じて、プログラミングの知識が少なくともアプリケーションの開発を実行することができることを身をもって感じる事ができた。

今後の課題は、改良点としては、線の太さや種類、フロントカメラ、バックカメラの切り替え機能の追加、スタンプ機能を追加し、よりユーザーを楽しませることができると考えている。

[参考文献]

[1]細谷日出海：“iPhone デジカメプログラミング”

ソフトバンククリエイティブ株式会社 P23-33,P66-106 ,2011年3月31日初版発行

[2]TJ 日記-iPhone アプリ開発などなど- : <http://blog.tapjockey.com/>