眼鏡を実装する108÷36の方法 [Rev.2]

ぽな@ばぐとら / SSP BUGTRAQ

はじめに:眼鏡の魅力について

諸君　私は眼鏡が好きだ　諸君　私は眼鏡が好きだ　諸君　私は眼鏡が大好きだ
あまりにも好きすぎて眼鏡の実装研究に明け暮れるとある開発者の妄想をお届けします。
今回も実装技術詳細の解説の都合上、伺「的」と言いながらSSP縛りになってしまうことをお許し下さい。

1.初級者向け:画像合成

- ふつうにoverlay+抜き色で合成
  いちばん簡単、ただしガタガタ
- 半透明情報を使用して合成(SSP限定+Windows2000以降限定)
  「眼鏡レンズ表面での反射」を表現
  抜き色黒塗りオプション(seriko.paint_transparent_region_black)をカット

2.中級者向け:着せ替え定義

- descript.txt内部の静的定義で実装
- 「グループ名」+「名前」の定義(sakura.bindgroup)　と　メニュー定義(sakura.menuitem)
- surfaces.txtのアニメーションIDと同じIDで定義しなければならないのに注意
- 通常同じグループ名の中ではどれか一つか、全く選ばないかの動きになる
  オプション指定で調整可能(複数自由選択や必ずどれか一つを選ぶオプション)
- メニュー定義をしなければ「スクリプトからのみ制御する着せ替え実装」が可能
  まったくメニュー定義がない場合は現状エラー扱い
- New! sakura.menu,autoと定義するとメニュー自動構築

3.上級者向け:ゴースト「the "GlassMaster"」

- 実は当たり判定は「外側」から取れる:Sakura APIや、SSTP Executeによる
- 顔(Face)当たり判定が取れれば、眼の位置はだいたい決定可能
  幸いたいていのキャラに定義されている
  デフォルメ顔が多い二次元キャラに合わせてだいたいの位置を調整
  (微妙にずれるけど)
- 要新規実装:「表情が切り替わったタイミングの通知の仕組み」
  OnOtherSurfaceChange
- 問題点:正面向き顔以外への対応が自動でできない
  真横向きになると大変なことに!

```
EXECUTE SSTP/1.4
Charset: Shift_JIS
Command: GetCollision
Sender: SSP
```

```
SSTP/1.4 200 OK

Head: 10,10,100,100
Face: 100,100,200,200
Bust: 200,200,300,300
```

4.超級者向け:すべてに眼鏡を!

- 大妄想大会
- OpenCVと呼ばれる画像処理・画像認識ライブラリを使う
  この中の機能を組み合わせ「画像中の特定の特徴を学習したデータをあらかじめ作っておき、それを利用して何かを自動認識する仕組み」がある
- 「正解画像」「不正解画像」を大量に集めて読み込ませ学習させることで、「顔」「手」などといった決まった特徴のある部分を認識できる。(何千枚レベルが必要なのでかなり泣ける)
- もちろん「アニメ顔」も認識可能(必要に応じて眼のみを再度探索も可能)
  実装例:http://www.ric.hi-ho.ne.jp/~woodvillage/program/AAMFaceAlign.htm
- 一度「顔」を認識したあと、その中から改めて「目」を認識で精度を上げる
- さらに発展編:顔以外の当たり判定全部も自動認識できないか?
- 誰かやってみませんか。