

# 極楽画像計測

(measure, merge, visualize)

## 取扱説明書

Ver. 2.2用 2022.07.05

鈴鹿工業高等専門学校 機械工学科

白井 達也

(shirai@mech.suzuka-ct.ac.jp)

## 【概要】

デジタルカメラで撮影した静止画やビデオカメラで撮影した動画から生成した静止画像（jpeg, png, tiff, tga）上をマウスでクリックして，その画像上のx, y座標をテキストファイルに記録するソフトウェアです． 1 画像あたり 1 個のテキストファイルが作成されます． Processing3 上で動作します（Processing3.5.4でのみ動作確認）． WindowsやLinux上で動作する実行形式も生成可能です．

ともかく大量の画像データの画像計測を素早く行いたい人向けです．

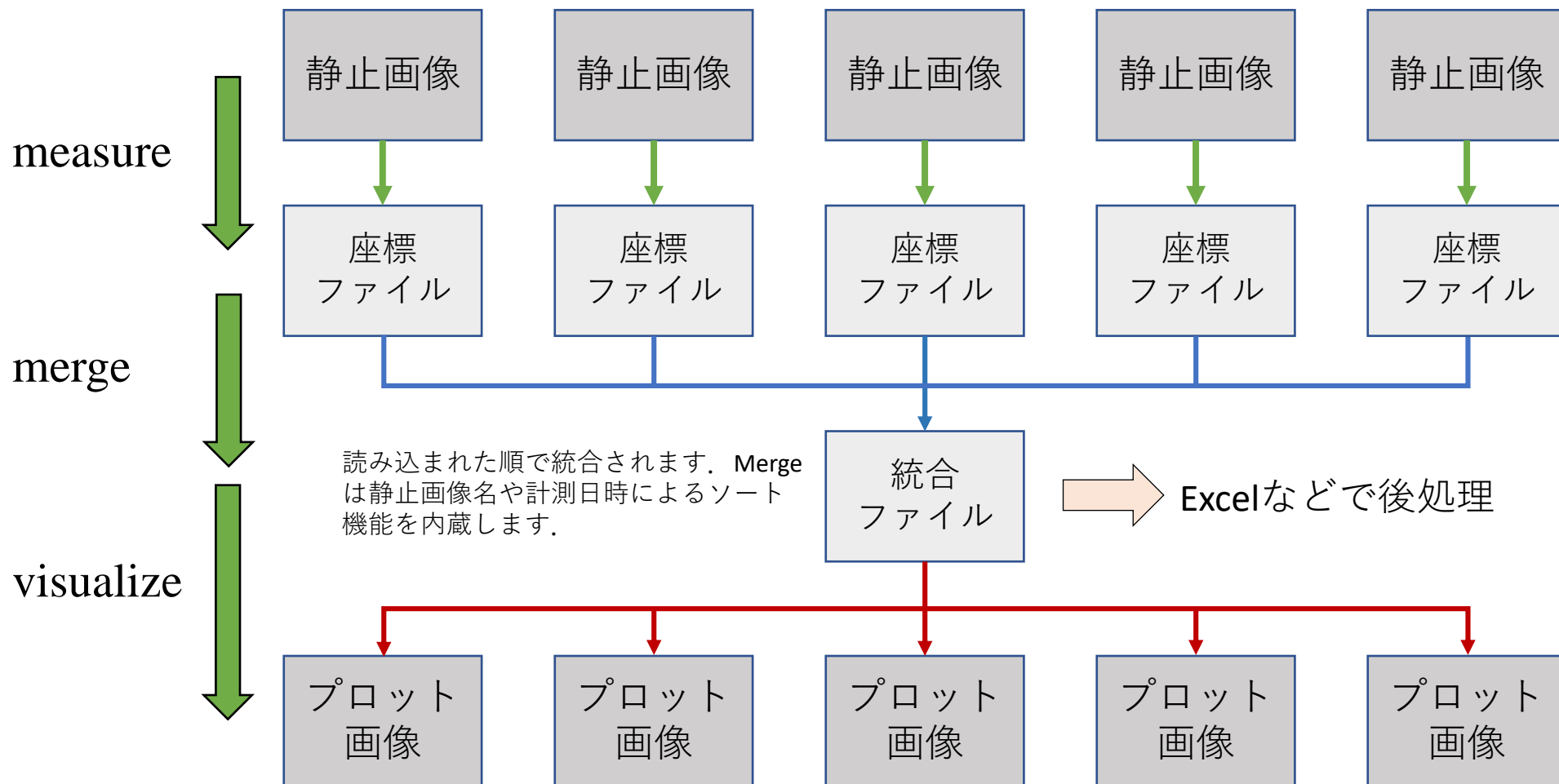
3つのソフトウェアで構成されます．

1. Gokuraku measure   ： 静止画像上の座標値をテキストファイルに記録
2. Gokuraku merge     ： measureで出力した複数のテキストファイルを一つに統合
3. Gokuraku visualize   ： 統合されたテキストファイルを画像に変換

measureで計測，それをmergeして，visualizeする． そういう流れです．

## 【イメージ】

静止画像 : オリジナルの画像（単発の静止画像，動画から生成した連番静止画像）  
座標データ : 計測して得られた点群のX,Y座標と色情報（画像データファイル名+“.txt”）  
統合データ : 座標データを1ファイルにまとめたもの（merged.txt）  
プロット画像 : 統合データから生成したプロット画像（merged-?????.jpg）



※プロット画像にはオリジナルの静止画像を合成可能です。

Stick pictureの描画にも対応（線分定義ファイルの読み込み）しています。

Processing IDE上のムービーメーカーでMOV(QuickTime)形式に動画に合成可能ですが，MOV型式の動画を再生できる環境は限られていますので注意が必要です（後述）。

## 【利用目的】

- 一般の方向けのソフトウェアではありません．主に研究者（学生含む）を対象としたソフトウェアです．
- モーションキャプチャーのシステムを持っている方には不要なソフトウェアです．
- カメラ撮影した静止画や，動画を何らかのソフトウェア（aviUtil, ffmpegなど）によって連続する静止画に分解したものの中の特徴点（事前に付けたマーカーなど）を次々とクリックして座標値をテキストファイルに記録するソフトウェアです．  
たとえば，スポーツや日常生活におけるヒトの体や指の動きの計測，移動ロボットの軌跡や姿勢の変化，流体中の物体の動き，大きく変形する材料の歪み，植物の成長，マウスや昆虫などの移動など，さまざまな用途が考えられます．
- 同様の機能を持つソフトウェア（製品／フリーウェア，例えばImage PCなど）も存在しますが，機能を限定しているため「点の座標だけを計測」する用途においては効率的かつ使い勝手が良いでしょう．
- リアルタイムに，動画中のマーカーの動きを計測するソフトウェアを産業用カメラメーカーなどが販売しています．静止画に変換する必要がありませんし，自動的にマーカーを追跡（Tracking）してくれるので，本ソフトウェアよりも高速かつ高精度でしょう．あくまで**安価に手軽に画像計測を行いたい層**がターゲットです．

## 【できないこと】

このソフトウェアでは2次元の画像から“画像上のx, y座標”(pixel)を取得することしかできません。それを実世界の座標系および単位 ([m], [mm]など) に変換したり, カメラの歪みを校正したり, ステレオカメラ画像から三次元座標をDLT法 (Digital Linear Transformation) などで取得する作業は別のソフトウェアを用いて下さい。

- 動画を連続する静止画像に変換する機能はありません。別のソフトウェア (AviUtil, Ffmpegなど) を使用して下さい。
- 特徴点を自動的に検出する機能はありません。あなたの目と指と根気が必要です。
- Visualizeでは人の全身のような複雑なStick pictureを描く機能もありますが, 各計測点間の関係を記述する便利なユーザーインターフェースはありません。線分定義ファイルをテキストエディターなどで作成して読み込んで下さい。
- 点と点の間を結んで線分とし, その線分の長さや線分間の角度を計測する機能もありません。これは各人, Excelやプログラミングで実現して下さい。あくまで各点の座標を計測する機能のみです (ただし, measureでは計測点の色情報を記録できますし, mergeでの統合も可能です。用途は想像できませんが) 。

## 【measure, merge, visualize共通の説明】

- measureで読み込める静止画像はJPEG, PNG, GIF, TGA形式, visualizeで出力可能なプロット画像はJPEG, PNG, TIFF, TGA形式です. measure, mergeで読み込み可能なテキストファイルはタブ区切りCSV形式です. Excel等を用いた手作業での修正も可能です.
- mergeで読み込む連続する静止画像のファイル名は「hoge hoge0000.jpg」のように拡張子の前に半角数字の付いた形式が好ましいです（勿論, 自由なファイル名でも問題はありません）. measure, mergeで複数のファイルを一度に読み込む際, ファイル名でソーティングしてからファイルを読み込みます.
- プログラムの終了は[ESC]キーです.

（技術的な補足）

measure, merge, visualizeは, フォルダー選択およびD&Dにおいて, ファイルの大雑把な指定（無関係なファイルも含む選択対象の中から, 対応するファイルのみを抜き出して読み込む）が可能です. ただそのファイルが対応するファイルかどうかを調べるために, ファイル全体を読み込む必要があります. もしその候補のファイルが動画ファイルだった場合は, 数十MBのファイルを一旦読み込んでしまいます. この無駄を省くために, ファイルの拡張子から明らかに無関係だと判明するファイルは読み込むことなく候補から外します.

common.pde中の以下の二つの文字列配列で指定しています.

```
final String [] extensionListOfImage = {"jpg", "jpeg", "gif", "tga", "png"};
```

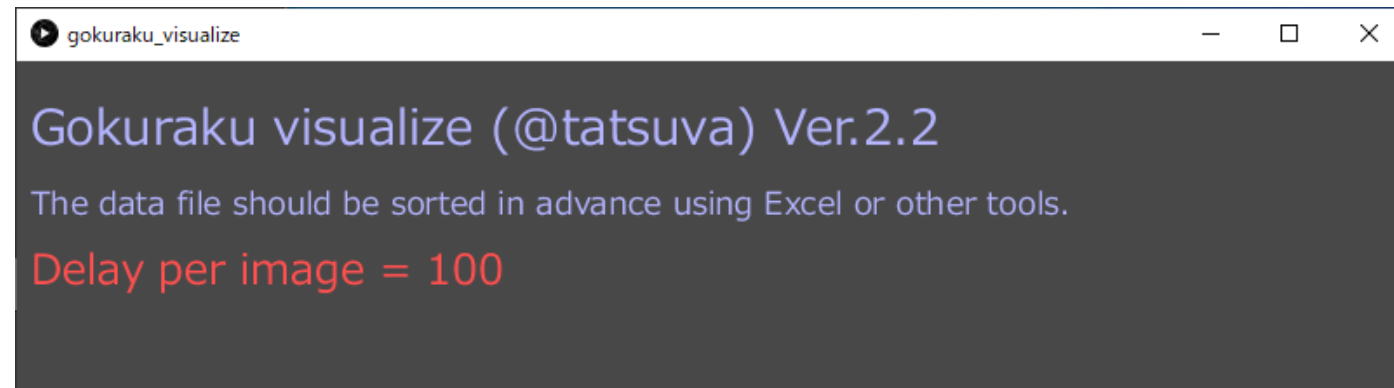
```
final String [] extensionListOfText = {"txt"};
```

- measure : extensionListOfImage[]内の拡張子を持つファイルを静止画像として読み込む
- Merge : extensionListOfText[]内の拡張子を持つファイルを座標データとして読み込む
- visualize : extensionListOfText[]内の拡張子を持つテキストを統合ファイルとして読み込む.

実際に読み込めるかどうかは, 拡張子による判定後に改めて行われます.

## 【measure, merge, visualize共通の説明】

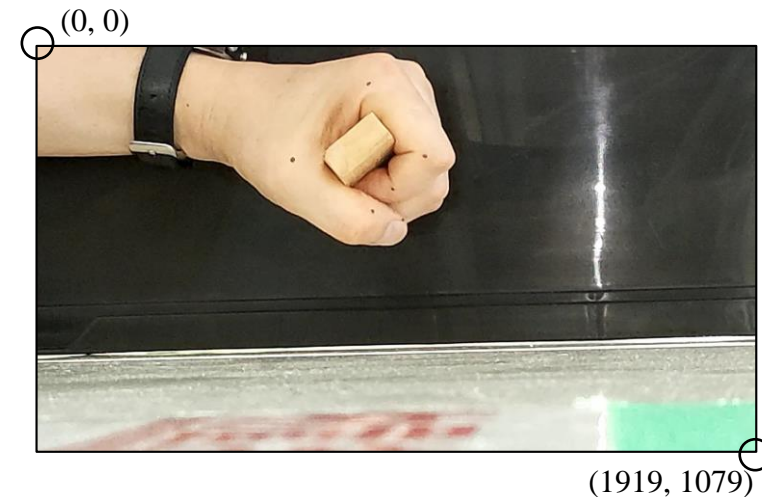
- ソースリストは分割されています. `common.pde`は3つのプログラム共通のコード(完全に同一内容)です. `lib.pde`は各プログラム専用のコードです.
- 各種設定の初期値は`header.pde`に記述してあります. 変数名は大文字から始まります.
- `measure`と`visualize`は各種設定を変更可能なサブメニューが存在します. `merge`は読み込んだ統合データのソーティングを行うサブメニューを持ちます.
- いくつかのパラメーターはキーボード入力可能です. 入力開始時には現在の値がプリセットされます. `[BackSpace]`キーにより1文字単位で消えますが, `[SPACE]`バーを押すと0に初期化されますので, 値を入力する前に`[SPACE]`バーを押すと変更が楽です. `[Enter]`キーを押すと確定します. `[ESC]`キーを押すと入力作業を中断できます. 入力して変更した設定値はプログラムを終了すると消えますので, 毎回, 同じ値に設定したい場合は`header.pde`内の初期値を書き換えるといいでしょう.



## 【measure, merge, visualize共通の説明】

### (座標系)

- 座標系は静止画像の左上が(0, 0)です.
- 右の画像は1920×1080[pixel]の画像です (各ファイルのプロパティの“詳細”タブを確認下さい). この場合, 右下の座標は(1919, 1079)です.



### (Y軸を反転したい) : (注) 変更するとvisualizeできません

- 計測してmergeしたデータをExcel等で処理する際に, 元画像の縦のピクセル数-1 (例では1079) からY座標の値を引けば, 左下が(0, 0)の座標系 (単位はpixel) に変換できます.

### (画像の傾きを補正したい) : (注) 変更するとvisualizeで静止画像と合成した際にズレます

- 回転行列は  $\begin{bmatrix} P'_x \\ P'_y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} P_x \\ P_y \end{bmatrix}$  です. 原点を基準にして角度 $\theta$ 回転するには,  $P'_x = P_x \cos\theta - P_y \sin\theta$ ,  $P'_y = P_x \sin\theta + P_y \cos\theta$ を計算して下さい (Excelの三角関数の引数はラジアンですのでご注意ください).

### (画像の比率を換算したい) : (注) 変更すると静止画像と合成した際にズレます.

- 画像中に既知の長さの物体 (方眼紙や定規) を入れて撮影し, その中の基準の長さを計測します. 例えば長さ10[mm]の物体が120pixelだったら, 1/12 [mm/pixel]です. この比率をx, y座標に掛ければ単位を[mm]に変換できます.
- カメラ撮影の特性上, 縦横比の微妙な違いや周辺部での歪みが発生する可能性があります.
- 奥行方向の歪みは遠方からズーム撮影することで, 影響を小さくすることができます.



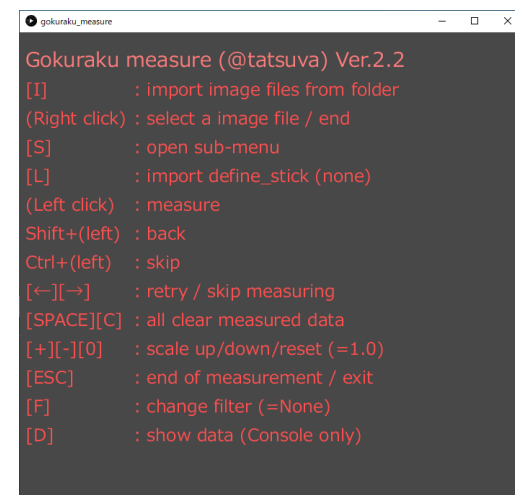
# 【Gokuraku measureの使い方】

## （静止画像の読み込み方）

単一選択，フォルダー選択，D & Dの三種類の方法があります．

- **単一選択**：メニュー画面を右クリックするとファイル選択ダイアログボックスが表示されます．ファイルを1個ずつ選択する場合です．
- **フォルダー選択**：メニュー画面で「I」キーを押して表示されるフォルダー選択ダイアログボックスで選択したフォルダー内の画像ファイルが計測対象となります．ただし，選択したフォルダーの下階層のフォルダー内の画像は計測対象になりません．
- **D & D（ドラッグ&ドロップ）**：エクスプローラーなどで複数のファイルを選択してメニュー画面にドロップします．ドロップしたファイル内にフォルダーが含まれていても，そのフォルダー内の画像は計測対象になりません．ただし，特例として単一のフォルダーのみをD & Dした場合，そのフォルダーに含まれる画像ファイルはフォルダー選択時と同様に計測対象になります．

※ フォルダー選択，D&Dで複数ファイルを同時に選択した場合，ファイル名で静止画像ファイルをソーティングしてから順次読み込みます．選択されたファイルが画像データではない場合は無視されますので，あまり気にせずに大雑把に放り込んで頂いてOKです．

A screenshot of the Gokuraku measure application window. The title bar shows 'gokuraku\_measure' and standard window controls. The main content area displays a list of commands and their functions in a monospaced font. The text is as follows:

```
Gokuraku measure (@tatsuva) Ver.2.2
[I]           : import image files from folder
(Right click) : select a image file / end
[S]           : open sub-menu
[L]           : import define_stick (none)
(Left click)  : measure
Shift+(left)  : back
Ctrl+(left)   : skip
[←][→]       : retry / skip measuring
[SPACE][C]    : all clear measured data
[+][−][0]     : scale up/down/reset (=1.0)
[ESC]         : end of measurement / exit
[F]           : change filter (=None)
[D]           : show data (Console only)
```

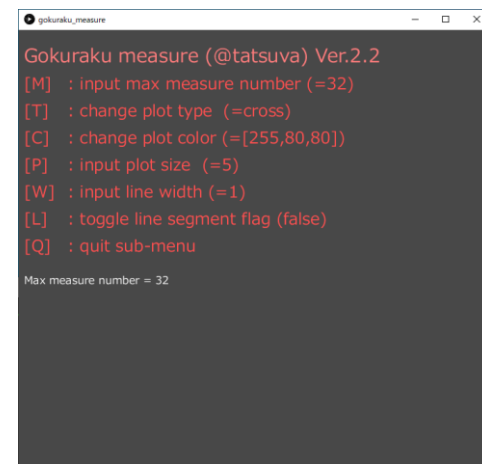
# 【Gokuraku measureの使い方】

## （各種設定の変更）

メインメニューで[S]キーを押すと設定変更のサブメニューに切り替わります。[Q]キーでメインメニューに戻ります。なお、計測中にもメインメニューやサブメニューの機能は利用できます（画像表示が優先されるので文字表示は見えませんが）。

- [M]キーを押すと最大計測点数（max measure number）を変更できます。初期値はheader.pdeのMax\_measureです。ここで指定された数の計測が終了したら自動的に計測を終了して座標ファイルを出力します。数値に0を指定すればこの機能を無効化できます。計測対象の静止画像群の測定数が同じ場合は、設定すると効率的に作業を進められます。
- [T], [C], [P], [W]キーを押すと、計測時のプロットの形状（noneはプロットなし）、色、サイズ、線幅を変更できます。色はheader.pdeでプリセットした色の組み合わせから選択されます。これらの初期値を変更したい場合は修正して下さい。
- [L]キーを押すと、プロット間を“線で結ぶ”／“結ばない”を切り替えられます。この機能はvisualizeの線分描画機能（Stick picture）と等価です。メインメニューで線分定義ファイルの読み込みも可能です。

※ 事前に最大計測点数を指定し、線分設定ファイルを読み込んでから計測を開始すると、ミスなくスムーズに測定が行えます。



## 【Gokuraku measureの使い方】

### （計測方法）：その前に

- 読み込んだ画像が小さ過ぎる／大き過ぎる場合は，[+]／[-]キーを押して拡大／縮小が可能です．[0]キーを押せば拡大率（**scale**）は1に戻ります．計測結果は拡大率を掛けた座標系で記録されます（画面上に表示された画像の座標）．計測時の表示上の拡大率は座標ファイルに記録され，**visualize**で利用可能です．
- 計測対象の静止画像が既に計測済みであった場合，自動的に記録時のスケールで静止画像と座標データが読み込まれます．もしこの仕様が気に入らない場合は，**header.pde**の**Force\_update**を**true**に変更して下さい（**Ver. 2.1**以前の仕様に戻る）．
- [L]キーを押すと，線分定義ファイルを読み込みます．詳しくは【**visualizeの使い方**】を参照して下さい．
- [F]キーを押すと，読み込む静止画像の色をグレースケール化，色反転可能です．
- 計測中に[D]キーを押すと，**Processing**の統合開発環境では画面下のコンソール領域に，現在の計測対象画像の計測済みのデータ（座標，色情報）が一覧表示されます．

## 【Gokuraku measureの使い方】

### （計測方法）

- 画像中で左クリックした位置の  $x, y$ 座標と色（ $R, G, B$ ）が計測されます．間違った場所をクリックした場合は，[Shift]キーを押しながら左クリックすると取り消せます．
- 記録したい特徴点が隠れているなどして計測できない場合は，[CTRL]キーを押しながら左クリックすると計測をスキップできます．その際の座標値は-1, -1, 色情報は0, 0, 0が座標ファイルに保存されます．
- [ESC]キーを押すか，右クリックするとそこで計測終了です．1ヵ所でも計測済みの場合，計測が終了すると座標ファイルは強制的に上書きされます．
- [SPACE]バーまたは[C]キーを押すと，記録された計測点が全てクリアされます．
- 計測した日時は座標ファイルにyyyymmddhhmmssの形式で保存されます．
- 静止画像が表示されている状態で[←]／[→]キーを押すと，一つ前（リトライ）／次の画像（スキップ）へ移動可能です．移動前に計測した計測点は破棄されます．
- リトライ／スキップで静止画像を移動した際に，既にその画像が計測済みであればプロットが表示されます．[Shift]+左クリックで一つずつ取り消したり，[SPACE]バーか[C]キーを押して全ての座標データを削除して計測し直すことも可能です（header.pdeのForce\_updateをtrueにすると，この機能は無効化できます）．

## 【Gokuraku measureの使い方】

### （座標ファイルのフォーマット）

- 1 行目：拡大率，画像ファイルの幅，画像ファイルの高さ，画像ファイル名，計測日時
- 2 行目以降：計測点，x座標，y座標，色情報（赤），色情報（緑），色情報（青）

- ・ 計測点は1番目が0，2番目が1と，実際の計測順番よりも1小さな値です．
- ・ 画像ファイルの高さと幅は拡大率と関係なく，オリジナルのサイズです（pixel）．
- ・ x, y座標は画像左上隅を(0, 0)とし，右方向，下方向を正とする座標です（pixel）．座標値は画面に表示されていた静止画像の生のx,y座標ですので，一連の静止画像の中で拡大率を変更すると座標値の倍率が一部だけ変わってしまいますので注意して下さい．visualizeは拡大率を無視して拡大率1相当に補正する機能がありますので，可視化においては問題はありませんが，統合データをvisualize以外のソフトウェアで加工する場合は，拡大率でx, y座標値を割るなどして正規化して下さい．
- ・ 計測をスキップした点の座標は(-1, -1)です．
- ・ 色情報(RGB)は0～255の値で，値が大きいほどその色成分が明るいことを意味します．

### （出力される座標ファイルについて）

- ・ 出力される座標ファイルはテキストファイルですので，テキストエディターでの編集やExcelなどでの読み込みが可能です．データ間はタブ記号により区切られます．
- ・ 出力される座標ファイルは画像データと同じフォルダーに保存されます．
- ・ 座標ファイルのファイル名は画像ファイルのファイル名に".txt"を追加した名前になります（例：human01.jpg -> human01.jpg.txt）

# 【Gokuraku mergeの使い方】

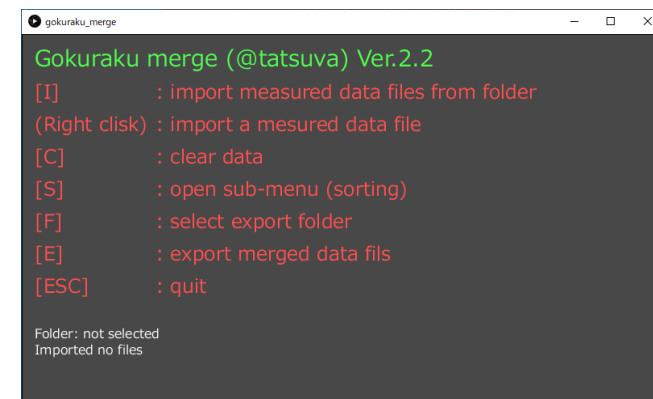
## （座標ファイルの読み込み方）

単一選択，フォルダー選択，D & Dの三種類の方法があります．

- **単一選択**：メニュー画面で右クリックするとファイル選択ダイアログボックスが表示されます．ファイルを1個ずつ選択する場合です．
- **フォルダー選択**：メニュー画面で「I」キーを押して表示されるフォルダー選択ダイアログボックスで選択したフォルダー内の座標ファイルが統合対象となります．ただし，選択したフォルダーの下階層のフォルダー内の計測データは統合対象になりません．
- **D & D（ドラッグ&ドロップ）**：エクスプローラーなどで複数のファイルを選択してメニュー画面にドロップします．ドロップしたファイル内にフォルダーが含まれていても，そのフォルダー内の座標ファイルは統合対象になりません．

座標ファイルを読み込むたびにデータが追加されていきます．[C]キーを押すと今までに読み込んだデータを全て破棄します．

※ フォルダー選択，D&Dで複数ファイルを同時に選択した場合，ファイル名で座標ファイルをソーティングしてから順次読み込みます．選択されたファイルが座標データではない場合は無視されますので，あまり気にせずに大雑把に放り込んで頂いてOKです．



# 【Gokuraku mergeの使い方】

## （ソーティングサブメニュー）

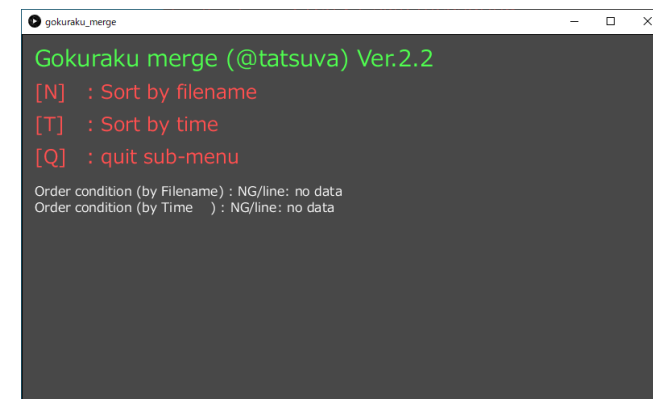
メインメニューで[S]キーを押すと、ソーティングサブメニューに切り替わります。

座標データは前ページの3種類の方法で読み込んだ順でメモリー内に蓄えられます。このメモリー内の座標データの並び順で統合ファイルは出力されます。ソーティングサブメニューでは、各座標データが静止画像ファイル名または計測日時の昇順で読み込まれているかをチェックする機能、静止画像ファイル名または計測日時に昇順にソーティングする機能を持ちます。

Order condition (by Filename), Order condition (by Time)がOKの場合は静止画像ファイル名、計測日時に昇順、NGの場合が昇順ではないことを表します。NGの場合のlineは問題の最初に発見された行番号（統合ファイル）です。OKの場合は読み込み済みの座標ファイル数です。

[F]キーを押すと静止画像のファイル名、[T]キーを押すと計測日時にメモリーに読み込まれた座標データをソーティングします。

※ 測定にミスがあることに後から気付いて特定のファイルを再測定した場合、他の座標ファイルに比べてこの座標ファイルの計測日時だけが大きく遅れているでしょう。ファイル名、日時にソーティングを繰り返しても、二つのOrder conditionが共にOKにはなりません（それで正しい）。どちらを優先するか、です（大半はファイル名）。



## 【Gokuraku mergeの使い方】

### （統合ファイルの出力）

- [F]キーを押すと統合ファイルを出力するフォルダーを選択可能です。なお、座標ファイルを読み込んだ際に、最後に読み込まれた座標ファイルのフォルダー名が自動的に設定されます。もし、どうしても別のフォルダーに出力したい場合のみ、エクスポート直前に変更して下さい。
- [E]キーを押すとメモリー中の統合された座標データを統合ファイルに出力します。ファイル名はheader.pde内で指定された“merged.txt”で固定です。変更したい場合はheader.pdeを修正して下さい。

※ 座標ファイル中の色情報（RGB）はデフォルトでは統合しません。もし色情報も統合したい場合はheader.pde内のMerge\_color\_info = falseをtrueに変更して下さい（使用頻度が低いのでメニューによる変更には対応しません）



## 【Gokuraku mergeの使い方】

### （統合ファイルのフォーマット）

全行以下のフォーマットです.

画像ファイル名, 計測日時, 拡大率, 画像ファイルの幅, 画像ファイルの高さ, 計測点数,  
1 個目の計測点のx座標, 1 個目の計測点のy座標, 2 個目の計測点の...の繰り返し.

### （座標データ読み込み時の注意点）

- mergeに読み込まれた順番で統合されます. フォルダー選択やD & Dの場合, 期待した順番で読み込まれない場合があります. 期待した順番で読み込まれたかどうかはエクスポートした統合ファイルの1列目の画像ファイル名か, 2列目の計測日時で判断できます. 統合ファイルをExcelなどに読み込み（タブ区切りCSV）, 1列目または2列目で“並べ替え”すると良いでしょう.
- （上記補足）Ver.2.2からは, 選択された複数の座標ファイルを読み込む際に, ファイル名でソーティングするようにしていますので, ほぼ（名前順による）期待通りの順番で読み込まれるでしょう. 不安でしたらソーティングサブメニューで確認, 出力した統合ファイルで確認して下さい.

# 【Gokuraku visualizeの使い方】

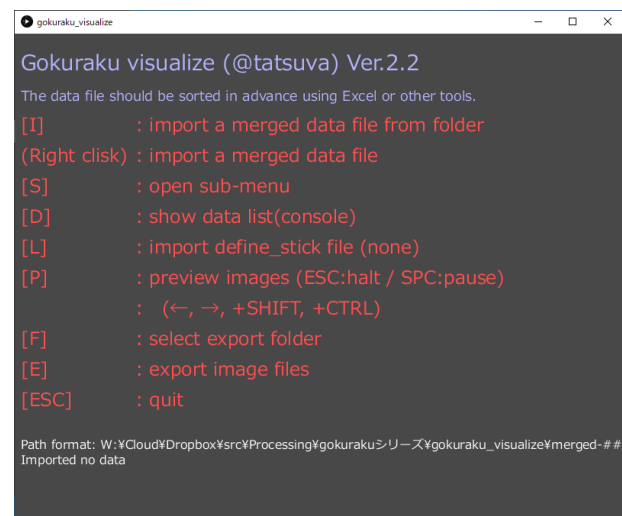
## （統合ファイルの読み込み方）

読み込むファイルは一つのみですが、フォルダー指定やD&Dで大雑把に指示することも可能としました。（大量の画像ファイルなどの中から単一選択は困難なため）。

- **単一選択**：メニュー画面で右クリックするとファイル選択ダイアログボックスが表示されます。統合ファイルを選んで下さい。
- **フォルダー選択**：メニュー画面で「I」キーを押して表示されるフォルダー選択ダイアログボックスで選択したフォルダー内の統合ファイルが読み込まれます。ただし、選択したフォルダーの下階層のフォルダー内の計測データは統合対象になりません。
- **D & D（ドラッグ&ドロップ）**：エクスプローラーなどで統合ファイルを選択してメニュー画面にドロップします。フォルダーをドロップしても構いません。

※ 選択されたファイルが統合データではない場合は無視されますので、あまり気にせずに大雑把にフォルダー選択あるいはD&Dで放り込んで頂いてOKです。ファイル名もチェックしていませんので、merge.txt以外のファイル名でも問題ありませんが、同時に選択された候補の中に複数の統合データファイルが存在した場合、最初にフォーマットが一致したファイルのみを読み込みますので、その点だけは注意して下さい。

※ mergeと異なり、visualizeは統合データファイルを読み込む度にメモリークリアを行いますので、明示的にメモリークリアを行うコマンドはありません。もし複数の統合ファイルを同時に読み込みたい場合は、何らかの手段（テキストエディターなど）で統合ファイルを合体したファイルを新たに作成して読み込ませて下さい。

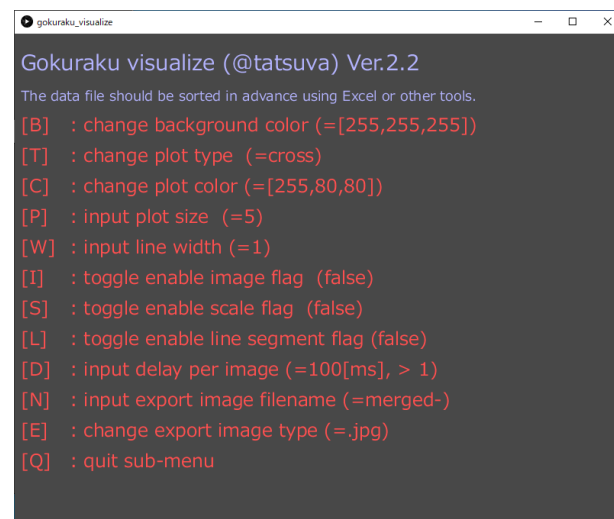


# 【Gokuraku visualizeの使い方】

## （各種設定の変更）

メインメニューで[S]キーを押すと設定変更のサブメニューに切り替わります。[Q]キーでメインメニューに戻ります。なお、計測中にもメインメニューやサブメニューの機能は利用できます。

- [B]キーを押すと、プロット画像の背景の色を切り替えられます。プリセットカラーは白、黒、灰色です。header.pdeを編集して変更できます。
- [T], [C], [P], [W]キーを押すと、計測時のプロットの形状（noneはプロットなし）、色、サイズ、線幅を変更できます。色はheader.pdeでプリセットした色の組み合わせから選択されます。これらの初期値を変更したい場合は修正して下さい。
- [I]キーを押すと、静止画像を“読み込まない”／“読み込む”を切り替えます。
- [S]キーを押すと、“計測時の拡大率で表示”／“元サイズで表示”を切り替えます。
- [L]キーを押すと、計測点間を結ぶ線分（line segment, Stick picture）を“表示する／しない”を切り替えます。デフォルトでは全計測点間を線で結びますが、後述するようにheader.pde内でDefine\_stick[][]を修正するか、線分定義ファイルを読み込むと、線分の組み合わせ、色、幅を細かく指定可能です。



# 【Gokuraku visualizeの使い方】

## (各種設定の変更)

- [D]キーを押すと、プレビュー時の画像間の遅延時間を変更できます。単位は[ms]です。
- [N]キーを押すと、出力する静止画像のファイル名を変更できます。使用できる文字はA-Z, a-z, 0-9, -() のみです。デフォルトは"merged-"で、header.pde内のImage\_filenameで設定されています。
- [E]キーを押すと、出力する静止画像の形式を変更できます。デフォルトはJPEGで、PNG, TIF, TGAに変更可能です。

- ◆ TIFF (Tag Image file Format)は可逆圧縮形式ですので画質の劣化がありません。その反面、ファイルサイズは小さくなりません。古典的な画像ファイル形式のため、多くのソフトウェアで読み書きが可能です。
- ◆ TGA (Truevision Graphics Adapter, TARGA)も古典的な画像ファイル形式ですが、TIFFに比べると対応しているソフトウェアは少なく、あえて選択するメリットはありません。
- ◆ JPEG (Joint Photographic Experts Group)は広く用いられる非可逆圧縮の画像ファイル形式です。圧縮率を高くすると画質が劣化してしまいます。visualizeではsaveFrame()コマンドでスクリーンを画像ファイル化していますが、圧縮率を指定できませんので、実質、どの程度の劣化なのかは不明です。もし気になるならばTIFF形式かPNG形式を選ぶと良いでしょう。
- ◆ PNG (Portable Network Graphics)はインターネットの普及に伴い広まった比較的新しい画像ファイル形式です。GIF形式の後継として開発され、可逆圧縮である点、透過色に対応するなど、JPEGよりも優れた点が多く存在します。

JPEG or PNGは圧縮率の違い、TIFF or PNGは対応ソフトウェアの違いで選ぶと良いでしょう。

## 【Gokuraku visualizeの使い方】

### （連続する静止画のプレビュー）

読み込んだ統合フォルダーと同じフォルダー内に座標を計測した際に用いた静止画像が存在する場合、画像ファイルを読み込んで、その上に計測点をプロットすることもできます（設定サブメニュー内の[I]コマンドで切り替え）。

- [P]キーを押すと先頭から順にプロット画像を生成して表示します。
- 連続再生中に[SHIFT]キーを押している間、再生速度は1/5に減速します。
- 連続再生中に[SPACE]バーを押すと一時停止、もう一度押すと連続再生に戻ります。
- 連続再生で最後のコマに到達したら一時停止に切り替わります。
- プレビュー中に[ESC]キーを押すとプレビューを中断します。

### （一時停止中のキー操作）

- カーソルキーの[→]キーを押せば一コマ進み，[←]キーを押せば一コマ戻ります。
- [→][←]キーと同時に[SHIFT]キーを押すと，最終のコマ，最初のコマにジャンプします。
- [→][←]キーと同時に[CTRL]キーを押すと，+ 1 0 コマ，- 1 0 コマ先に移動します。

※ プレビュー時の画像間の遅延時間は設定サブメニュー内で変更可能ですが，初期値はheader.pdeの中のDelay\_preview\_per\_imageで設定されています。速すぎたり遅すぎると感じる場合はこの数値を変更して下さい。

## 【Gokuraku visualizeの使い方】

### （連続する静止画のエクスポート）

デフォルトでは，連続する静止画像（プロット画像）ファイルは読み込んだ統合フォルダーと同じフォルダー内に保存されます．

- [F]キーを押すと，自動設定されたエクスポートフォルダーを変更できます．なお，統合ファイルを読み込むとそのフォルダーに変更されますので，必要な場合は都度変更して下さい．

- [E]キーを押すと，連続する静止画像をエクスポートフォルダーに出力します．

※この際の画面表示の速度はheader.pde内のDelay\_export\_per\_image, Delay\_export\_afterで変更できます．Processingの仕様上，この数値を小さくし過ぎると正常な画像出力が行われない恐れがありますので注意して下さい．

※ 出力される連続静止画像ファイルのファイル名はmerged-????.jpgです．????の部分は0000, 0001, 0002,...の順です．9999以上の場合は，統合ファイルの読み込み時に自動的に5桁以上に設定されますのでご安心ください．"merged-"の部分はheader.pde内のImage\_filenameを書き換えることで変更可能です．設定サブメニュー内でも一時的に変更可能です．

※ 技術的にはプレビューでスクリーンに描画した画像をsaveFrame()コマンドで画像に保存しているだけです．デフォルトではjpeg形式で保存しますが，tiff, tga, png形式でも保存可能です．保存形式は随時，設定サブメニューで変更可能です．header.pde内のImage\_type\_extの拡張子の順番を変更すれば，選択される優先順位を変更可能です．

## 【Gokuraku visualizeの使い方】

### (線分描画／Stick pictureの指定方法①：header.pde内の初期値Define\_stick)

形式： intの二次元配列です.

[Start, End, R, G, B, Width]

Start, End：線分を結ぶ計測点の番号です. 0から始まります（表示上は1から）.

R, G, B：色指定（0～255）

width：線の太さ（1以上, pixel）

例)

Define\_stick = {};（初期値）では全てのプロット間を計測順に線で結びます.

Define\_stick = {{0, 1, 255, 0, 0, 1}, {1, 2, 0, 255, 0, 2}, {2, 3, 0, 0, 255, 3}}; とすると,

計測点 0と1を赤, 太さ 1 で結ぶ.

計測点 1と2を緑, 太さ 2 で結ぶ,

計測点 2と3を青, 太さ 3 で結ぶ.

なお, 計測時にスキップした計測点 ([-1, -1]) は無視されます.

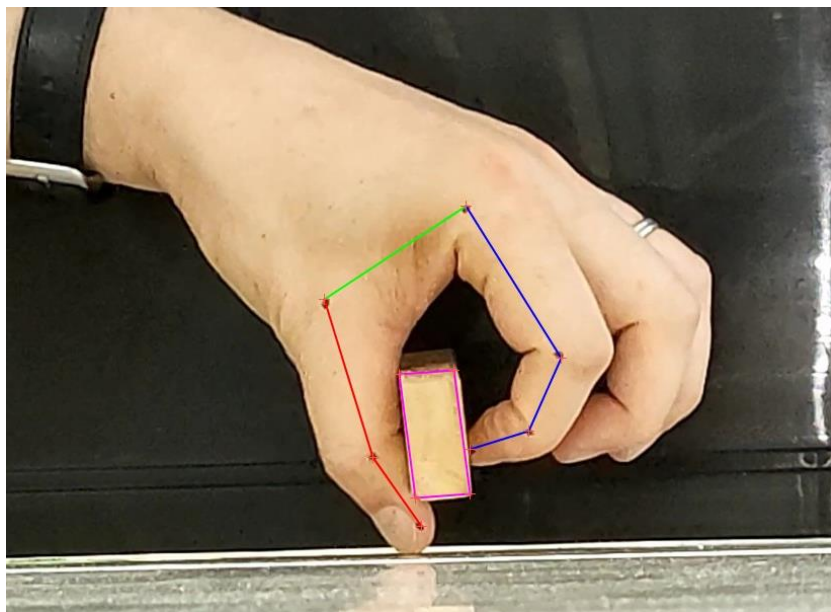
## 【Gokuraku visualizeの使い方】

### (線分描画／Stick pictureの指定方法②：線分定義ファイルのインポート)

- メインメニューで[L]キーを押すと，線分定義ファイルのインポートが可能です．
- 読み込む線分定義ファイルのファイル名は自由です．ファイルはタブ区切りCSVファイルで，

Start End R(0-255) G(0-255) B(0-255) Width  
の形式です．

※ 以下は，親指先端が計測点0，時計回りに1, 2までが赤，2と3の間が緑，3から人差し指先端の6までが青，四角柱断面の7～10までの4か所が紫，線幅2で指定した例です．



ファイル(F) 編集(E) 表示(V) 検索(S) ウィンドウ(W) マクロ(M) その他(O)					
define_stick.txt					
0	1	255	0	0	2↓
1	2	255	0	0	2↓
2	3	0	255	0	2↓
3	4	0	0	255	2↓
4	5	0	0	255	2↓
5	6	0	0	255	2↓
7	8	255	0	255	2↓
8	9	255	0	255	2↓
9	10	255	0	255	2↓
10	7	255	0	255	2↓
[EOF]					



# 【Gokurakuシリーズの実行環境構築】

## （動作確認済み環境）

Windows10 + Processing3.5.4 (64bit) です.

GokurakuシリーズはProcessing上で動作するように作成されています. パス区切りに関しては, MacOS/Linuxにも配慮して, 環境ごとに'¥'と'/'を使い分けるように作成しています. ただし, MacOS版/Linux版のProcessingでは動作確認を行っていません.

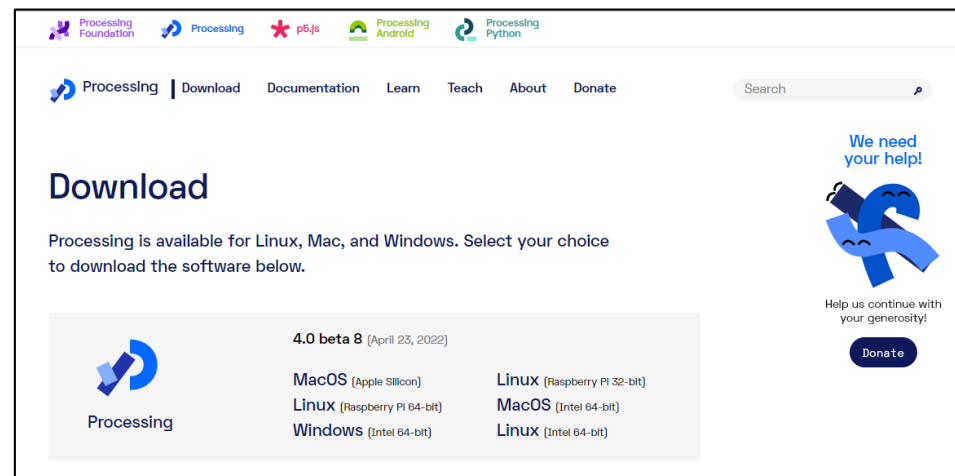
## （動作環境の構築）

Processingの統合開発環境（IDE）は, 以下のProcessing公式サイトダウンロードページからダウンロードできます. <https://processing.org/download/>

特にインストール作業は必要ありません. たとえばダウンロードしたZIPファイルをUSBメモリー上に解凍して, その上でProcessingの統合開発環境を実行しても動作します.

※ Ver.2.2から日本語フォントに対応しました（初期値はMeiryo）. 環境によっては選択したフォントを利用できないかも知れません. その場合はheader.pde内のJapaneseFontを変更して下さい.

※ Processingで読み込めても実行できない場合は, Java Runtime Environmentのバージョンが古い可能性があります. JREを更新してみてください.



# 【Gokurakuシリーズの実行環境構築】

## (最新版のGokurakuシリーズのダウンロードと実行)

GokurakuシリーズのソースリストとWindows版実行形式を同梱したZIPファイルは以下のOSDNのサイトからダウンロードできます。

<https://osdn.net/projects/gokurakumasure/releases/>

gokuraku\_measure-all-?.?.\*.zip に、measure, merge, visualize全てが含まれます。これらを好きなフォルダーに展開し、Processingから開けば実行可能です。

各フォルダー内にapplication.windows64というフォルダーがあります。その中のEXE形式のファイルを実行すれば、Processing無しでもWindows（64bit環境）上でGokurakuシリーズを実行できます。ただしEXEファイル単体では動作しません。その下にあるjava, lib, sourceのフォルダーのファイルも必要です。



## 【Gokurakuシリーズの活用例】 動画から連番静止画像を生成

### (動画の処理)

- AviUtil (KENくん作) : <http://spring-fragrance.mints.ne.jp/aviutil/>
- + 拡張編集プラグイン (KENくん作) : 同上 (exedit92.zip)
- + MP4入力プラグイン (muken氏作) : [https://pop.4-bit.jp/?page\\_id=7929](https://pop.4-bit.jp/?page_id=7929)
- + かんたんMP4出力 : <https://aoytsk.blog.jp/aviutil/34586383.html>
- + JPEG連番出力プラグイン (うえぽん氏作) : <http://auf.jpn.xxxxxxxx.jp/>

FFmpeg (<https://ffmpeg.org/>) を用いる方法もありますが、ここではAviUtil + プラグインを組み合わせての方法で説明します。まずAviUtilをダウンロードしてインストールします (AviUtil110.zipをダウンロードして適当なフォルダーに展開する)。aviutil.exeをダブルクリックして正常に起動することを確認後に、終了して下さい。

次に、aviutil.exeの存在するフォルダーの下に、Pluginsという名前のフォルダーを作成して、L-SMASH\_Works\_r940\_plugins.zipの中から、lwcolor.auc, lwdumper.auf, lwinput.aui, lwmuxer.aufの4つのファイルをコピーします ([参考URL](#))。

AviUtilを起動して、ファイル→環境設定→入力プラグインの優先度の設定で、右の図のようにL-SMASH Works File Readerが存在すればOKです。これでMP4形式の動画を読み込みます。



## 【Gokurakuシリーズの活用例】 動画から連番静止画像を生成

- 続けて、連番JPEG画像出力のための「JPEG3点セット」をPluginsフォルダーに展開します。（jpeg\_input.aui, jpeg\_output.auo, jpeg\_print.auf）  
ファイル→プラグイン出力に“連番JPEG出力（YC48）”が出現すれば正常に読み込まれています。
- かんたんMP4出力は、easym4.auoをPluginsフォルダーにコピーします。
- 拡張編集プラグインは、Pluginsフォルダーではなく、aviutil.exeと同じフォルダー内にZIPファイル内の全ファイルを展開して下さい。



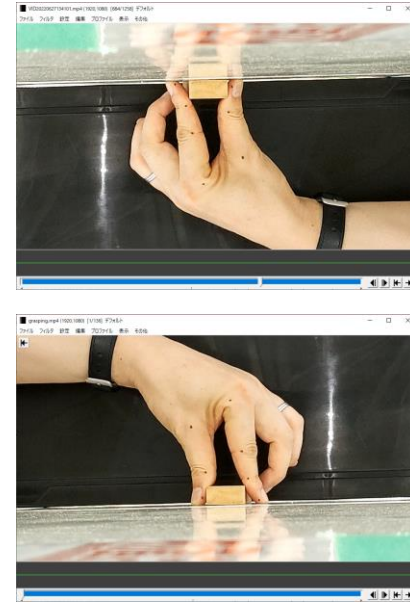
## 【Gokurakuシリーズの活用例】動画から連番静止画像を生成

スマートフォンで動画を撮影しました。AviUtilで読み込むと、上下が逆さまでした。スマートフォンを逆さまにして撮影した影響かも知れません（Windowsのプレイヤーなどで表示すると正常ですが）。

VideoStudio2021で必要なシーンのみ切り取って再レンダリングして出力したところ、正しい向きに直りました。撮影時に気を付けるか、別のソフトウェアで修正が必要かも知れません。

正常に読み込めたら出力しましょう。ファイル→プラグイン出力から、“連番JPEG出力（YC48）”を選択します。ファイル選択ダイアログボックスが表示されますので、ファイル名に例えば“grasping”のようなファイル名を入力して保存を押します。入力したファイル名に“-0000”を追加した大量のJPEGファイルが指定したフォルダーに生成されます。あとは、この静止画のフォルダーをmeasureで選択すれば計測開始です。

今回の例では約4秒の動画ですので、秒間30コマで136枚の静止画が生成されました。





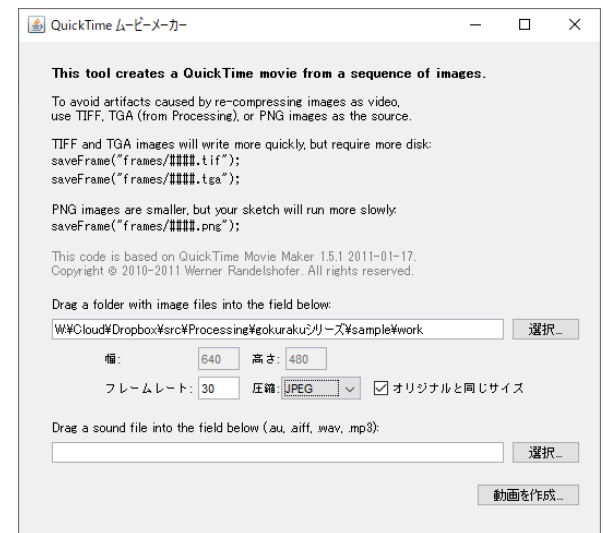
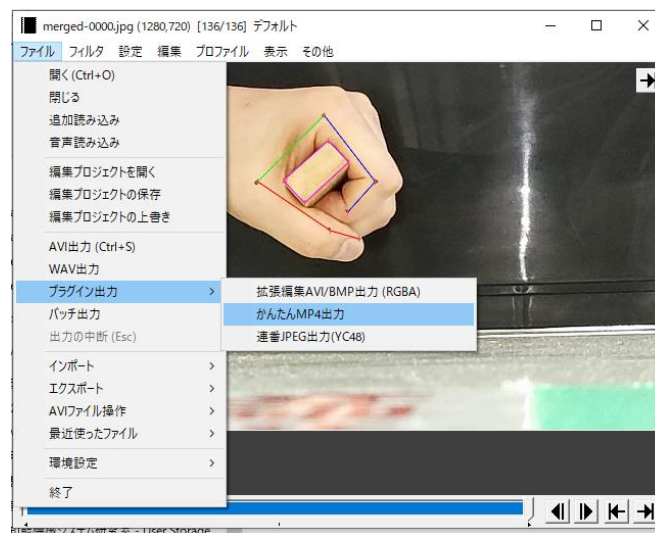
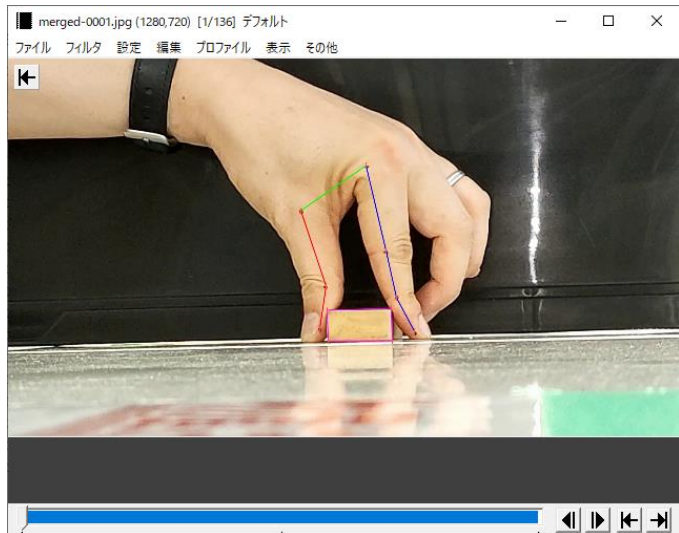
## 【Gokurakuシリーズの活用例】連番静止画像を読み込み，MP4形式の動画で出力

measure, merge, visualizeで処理を進めると，計測点，Stick picture，元画像を組み合わせた連番静止画像が得られます．これらをAviUtilに読み込みます．

visualizeはデフォルトでmerged-0000.jpgから静止画像を生成します．AviUtilでこの静止画像を読み込むと，自動的に残りの連番画像の番号順に読み込んでくれます．

以上で画像の連結が完了です．ファイル→プラグイン出力から“かんたんMP4出力”を選択してファイルを保存すれば完成です．

※ AviUtilではなく，Processing IDEのツール→ムービーメーカーでも連番静止画像（tif, tga, jpeg, png）からも動画を生成可能です．ただし生成される動画はMOV(QuickTime)形式です．圧縮形式でAnimationを選ぶとWindowsでは再生できないMOVファイルが生成されますが，JPEGを選べば再生可能です．ただ，手軽さを考えるとムービーメーカーも良いのですが，流通性を考えるとAviUtilによるMP4形式がお勧めです．なお，画質の指定が可能である点から，精度を重視する場合はFfmpegも検討すべきでしょう（本ドキュメントでは未紹介）．



## 【Gokurakuシリーズの活用例】 サンプルデータで動作確認

配布しているサンプルデータをダウンロードして展開して下さい。この中には、

- grasping.mp4 : 撮影した動画ファイル
  - grasping-0000.jpg～grasping-0135.jpg : AviUtilで分解した静止画像ファイル
  - grasping-0000.jpg.txt～grasping-0135.jpg.txt : measureで計測した座標ファイル
  - merged.txt : 上記座標ファイルをmergeで結合した結合ファイル
  - define\_stick.txt : 上記結合ファイル用の線分定義ファイル
  - merged-with-image.zip : visualizeで元画像ありで生成した画像ファイル
  - merged-without-image.zip : visualizeで元画像なしで生成した画像ファイル
- 
- measureでの計測点は親指先のマークから人差し指，四角柱の四隅の計11点（max measure = 11），スケールは0.666...です（縮小1回）。
  - visualizeでの画像生成はscale有効，define\_stickの読み込み（define\_stick.txt）です。

まず， grasping-?????.jpg.txtとmerged.txtを削除して下さい。それからmeasureで拡大率を0.666...に1回縮小し， max\_measureを11に設定， サンプルデータのフォルダーをmeasureのスクリーンにD&Dして計測開始。計測が終わったらmergeを起動し， 同じくサンプルデータのフォルダーをD&Dしてエクスポート。最後はvisualizeでmerged.txtをインポートすればプレビュー可能になります。