

# 数理と実装 2015 (10月3日)

# 私の実装

- Real-Time Human Pose Recognition in Parts from Single Depth Images (CVPR 2011 Best Paper)
- X-Box 360 の Kinect V1 の元になっている手法



## 簡単なおさらい(1/3)

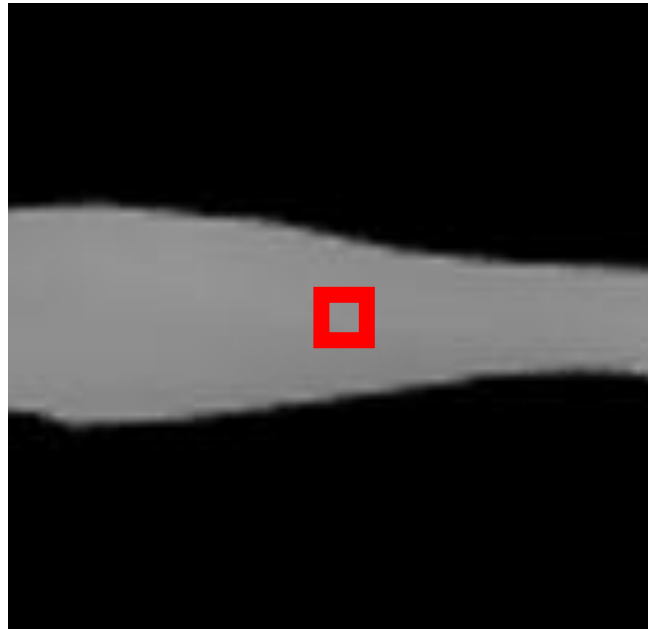
- デプスマップから各ピクセルでセグメンテーションを行います



# 原理説明 (超簡単版)

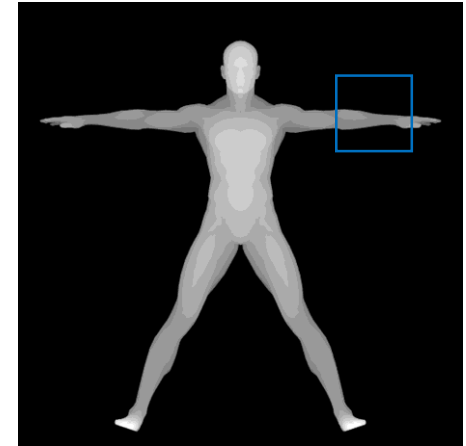
# 原理説明(超簡易版)

- クイズ: このデプスはどの部位でしょうか



# 原理説明(超簡易版)

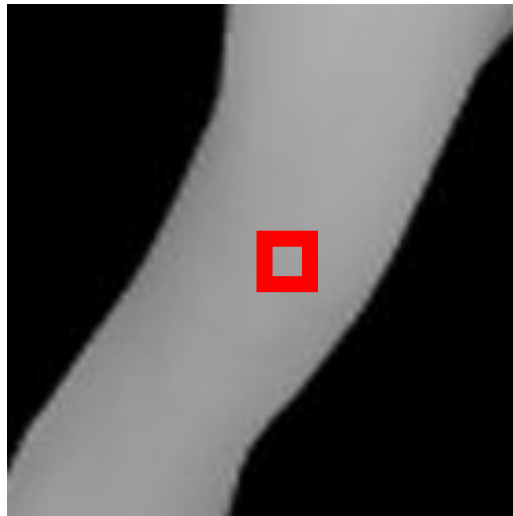
- クイズ: このデプスはどの部位でしょうか



右腕!!

# クイズ

- クイズ: このデプスはどの部位でしょうか



# クイズ

- このデプスはどの部位でしょうか



左足!! !

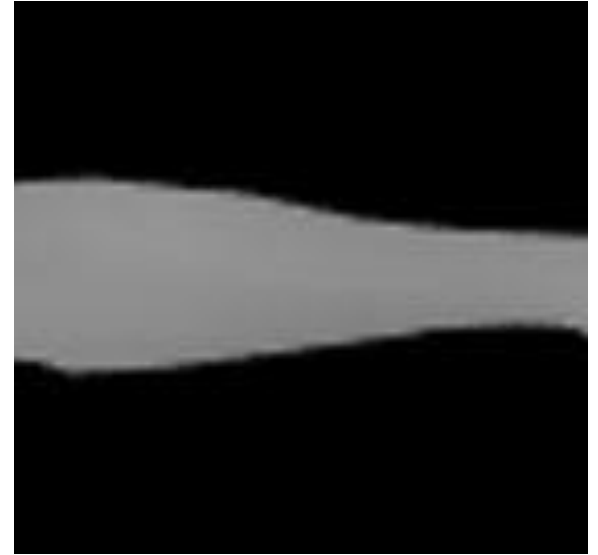




人間は経験則から、切り取られた Window から、どの部位かは推測出来る

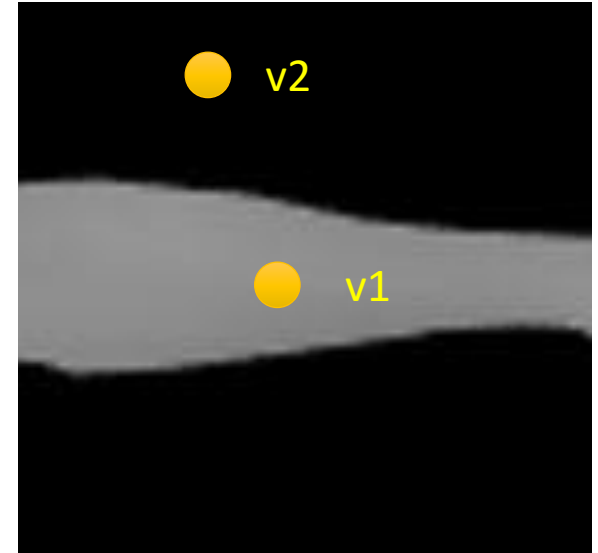


# アルゴリズムにしてみよう(2/3)



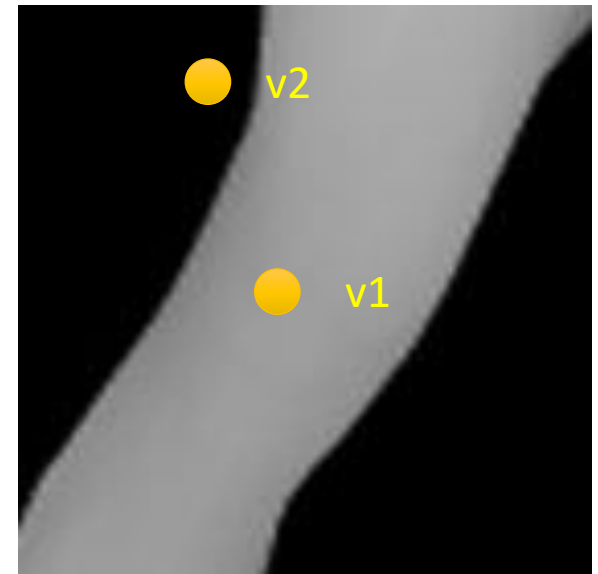
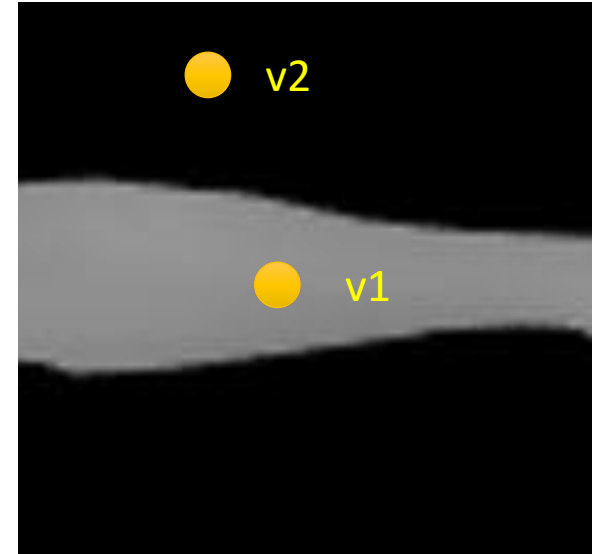
## アルゴリズムにしてみよう(2/3)

if(デプス(v1) - デプス(v2) > 20cm)  
 右手？



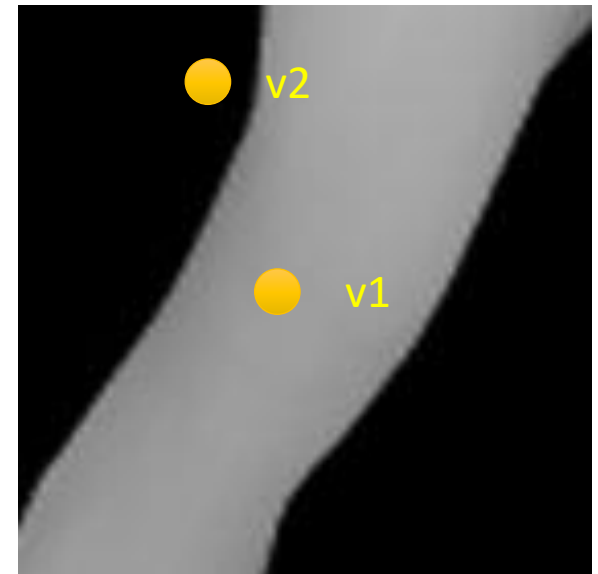
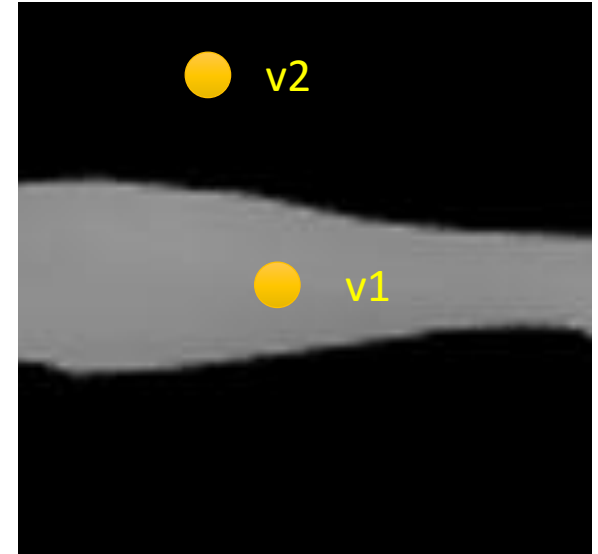
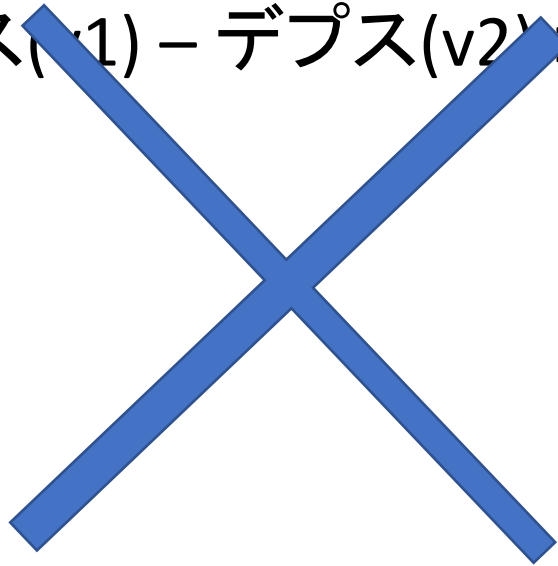
## アルゴリズムにしてみよう(2/3)

if(デプス(v1) - デプス(v2) > 20cm)  
右手？



## アルゴリズムにしてみよう(2/3)

if(デプス(v1) - デプス(v2) > 20cm)  
右手?



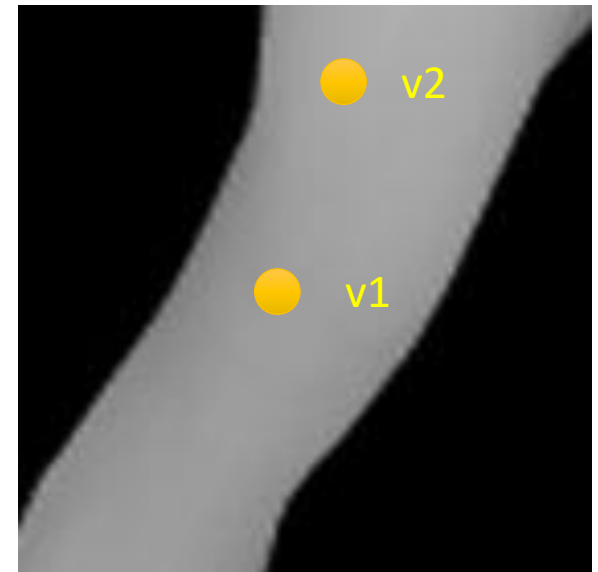
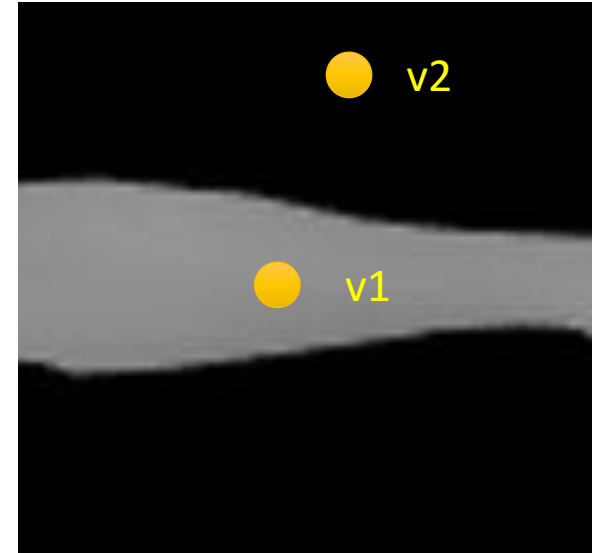
## アルゴリズムにしてみよう(2/3)

if(デプス(v1) - デプス(v2) > 20cm)

右手？

else

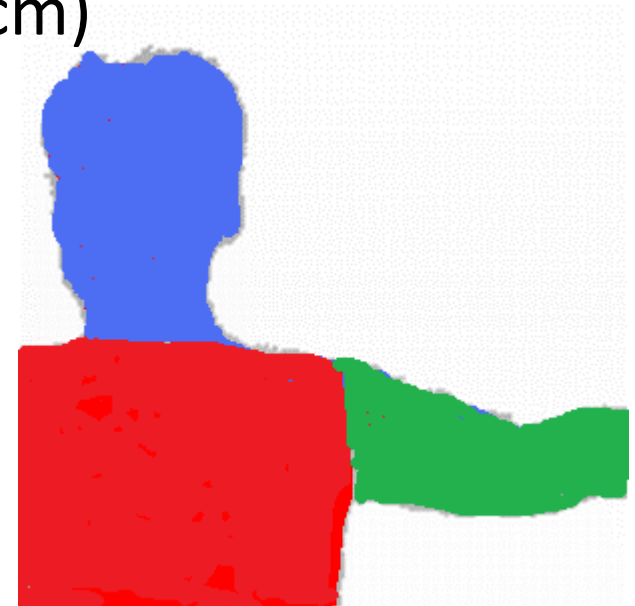
左手？



## 簡単なおさらい(2/3)

これを繰り返すと Kinect v1 で使われている、  
木構造の Decision Tree になります (神速！)

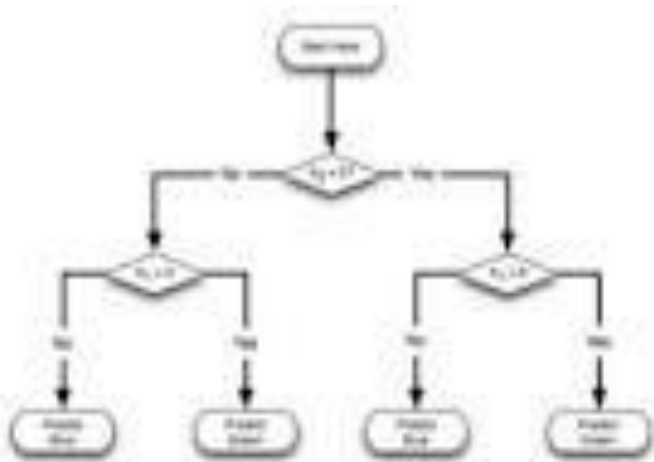
```
if(デプス(v1) – デプス(v2) > 20cm)  
  if(デプス(v3) – デプス(v4) > 11cm)  
    if(デプス(v5) – デプス(v6) > 32cm)  
      右手;  
    else  
      頭;
```



## 簡単なおさらい(2/3)

Decision Tree は弱識別機です

→ 極端な話、正解率 51 % とかでもいい。。。

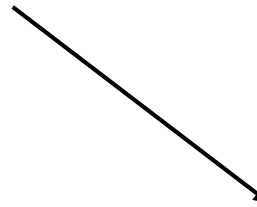




## 簡単なおさらい(3/3)

サンプルを適当に分割して、Decision Tree を作成して解決！ (弱識別器  $\times 100$  = 強識別器)

→ Random Forest



しかし、この研究のキモは、  
ランタイムのアルゴリズムで  
はない気がします

学習時にて、Decision Tree を作る  
際、正解データはどう用意する？

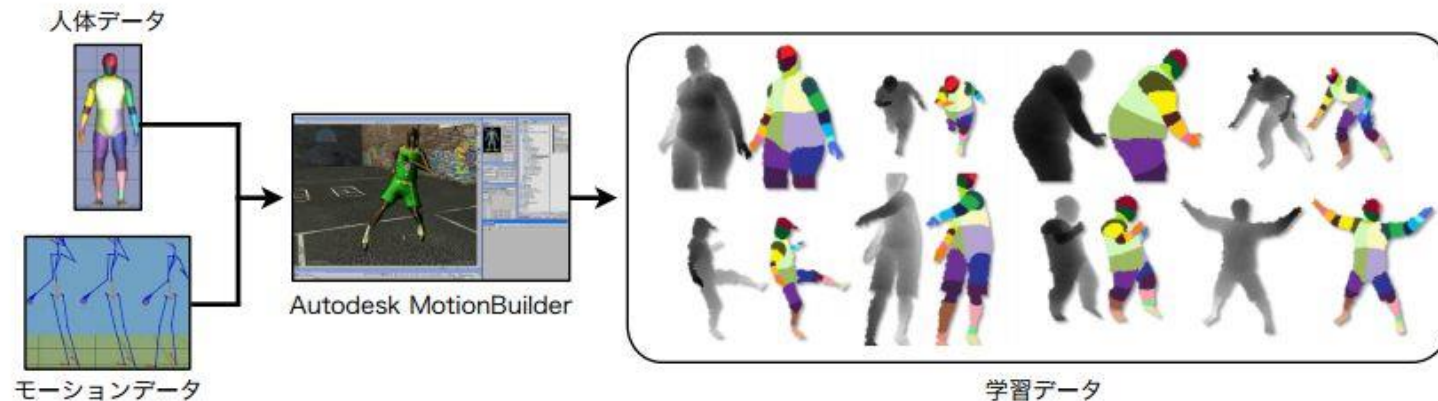


学習データ

# 学習データの自動生成

## ・ 学習データの生成の流れ

- Autodesk MotionBuilderより素体を選択
- 素体の各関節にモーションキャプチャのデータを設定し、3Dメッシュを変形
  - ・ 色々なパラメータをランダムに設定
    - 身長・幅・形状・カメラ姿勢・カメラノイズ・衣服・髪など
- 3Dメッシュをレンダリングし、デプスを取得



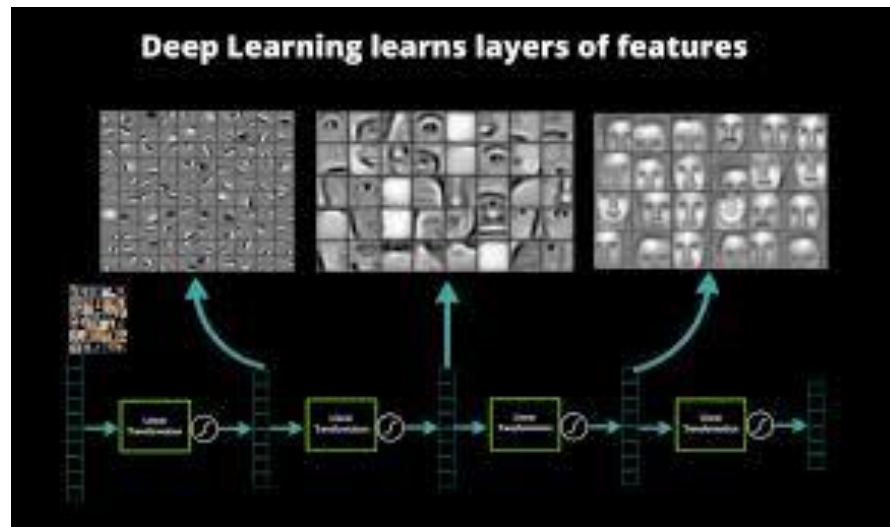
- もっと詳しい解説は Computer Graphics Gems JP 2015 に載ってます
  - ソースコードも git にあがってます
  - <https://github.com/DaidaYamamoto/KansaiCV/tree/master/BodyTrack>
  - 藤吉先生の「Random Forestsとその応用」の方が100倍分かりやすいです
- ~~デモしたかったんですが、Kinect 持ってくるの面倒だったです~~



ここから本番

- これだとただCVPR 論文実装しただけで終わってしまう
  - 製品向けのアルゴリズムなんで、そもそも新規性が薄い
  - 伸びしろの無さが凄い
- 他に応用例を考えてみる

# 世界中の人々が機械学習を連呼している時代



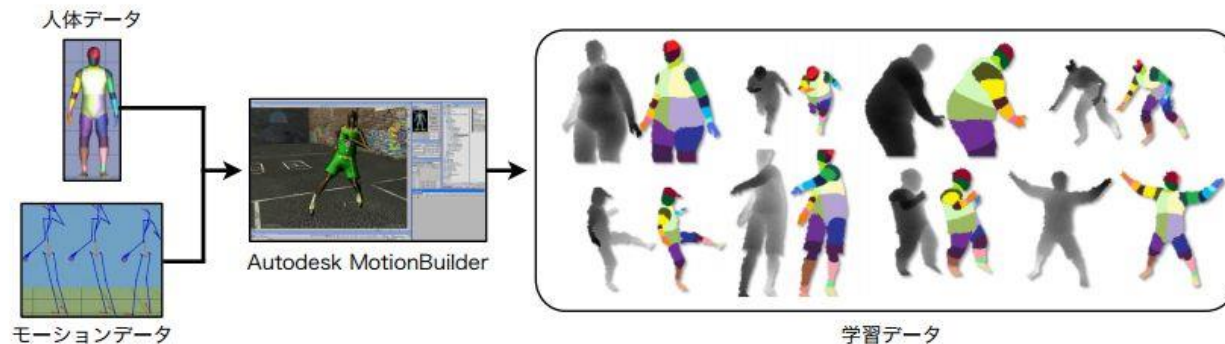
我々の強みを出すなら、今まで培ってきた CG の技法を使わないと出来ないことを考える



# Reminder.... (学習サンプルの生成)

## 学習データの自動生成

- ・ 学習データの生成の流れ
  - Autodesk MotionBuilderより素体を選択
  - 素体の各関節にモーションキャプチャのデータを設定し、3Dメッシュを変形
    - ・ 色々なパラメータをランダムに設定
      - 身長・幅・形状・カメラ姿勢・カメラノイズ・衣服・髪など
  - 3Dメッシュをレンダリングし、デプスを取得



# Reminder.... (学習サンプルの生成)

## • 手動タイプ

- イメージから物体識別

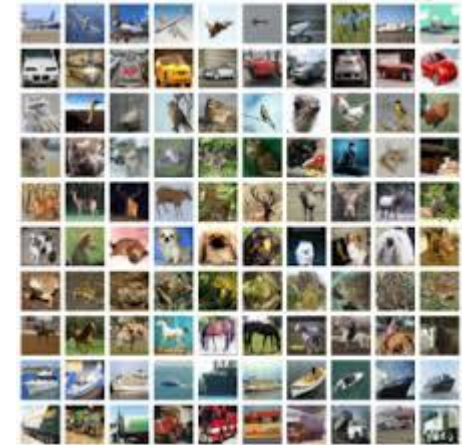
- CIFAR image classification benchmark

- 手書きの数字

- MNIST handwritten digits benchmark

- 顔のランドマーク

- 300 Faces in-the-Wild Challenge



## • 自動タイプ

- Real-Time Hand-Tracking with a Color Glove (Siggraph 2009)

- 手の 3D モデルを使って自動でアニメーションさせて、3Dレンダリング

- Real-Time Human Pose Recognition in Parts from Single Depth Images (CVPR 2011)

- モーションキャプチャのデータを人にあてはめて、3Dレンダリング

# (Procedural) Modeling + (PhotoRealistic) Rendering

- 顔
  - もう出来てる
    - Face Alignment at 3000 FPS via Regressing Local Binary Feature (CVPR 2014)
    - Caffeで試す絵師判定(SIG2D)
  - 製品化もしてる
    - OKAO, FACE++, Motion Portrait



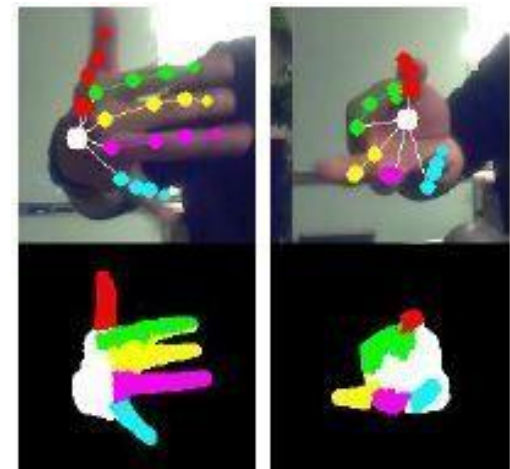
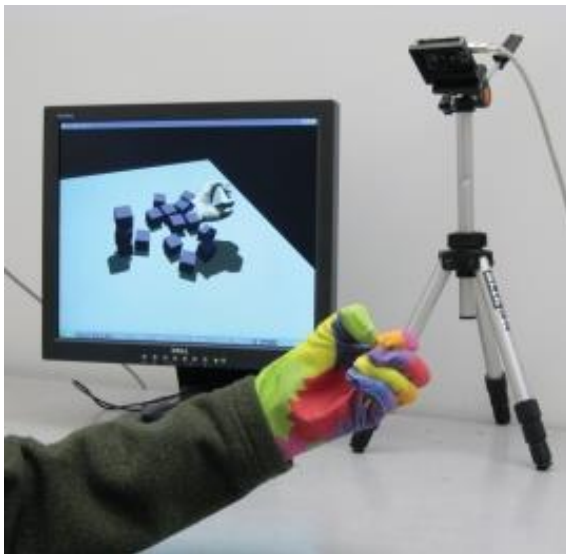
# •手

## •みんなやってる

- Cascaded Hand Pose Regression (CVPR 2015 Poster)
- Latent regression forest: Structured estimation of 3d articulated hand posture (CVPR 2014)
- Real-Time Hand-Tracking with a Color Glove (Siggraph 2009)

## •製品化もしてる

- Leap Motion, Kinect v2....



- 胴体
- みんなやってる
  - Real-Time Human Pose Recognition in Parts from Single Depth Images (CVPR 2011)
  - Accurate Realtime Full-body Motion Capture Using a Single Depth Camera (Siggraph Asia 2012)
  - Mosh: Motion and Shape Capture from Sparse Markers (Siggraph Asia 2014)
- 製品化もしてる
  - Kinect v2, Softkinetics





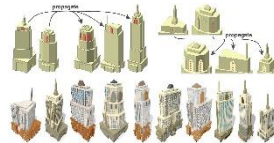
# CG の Procedural modeling ( Rendering ) (SG14/15 SGASIA 15 EG 15 から抜粋)

## • 建物

- Advanced Procedural Modeling of Architecture (SG15)
- Learning Shape Placements by Example (SG15)
- Interactive Dimensioning of Parametric Models (EG15)

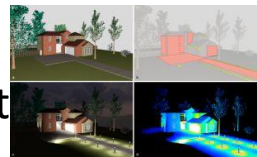
## • 林・木

- WorldBrush: Interactive Example-based Synthesis of Procedural Virtual Worlds (SG15)
- Controlling Procedural Modeling Programs with Stochastically-Ordered Sequential Monte Carlo (SG15)
- Inverse Procedural Modeling of Trees (EG14)



## • 外観

- Procedural Design of Exterior Lighting for Buildings with Complex Const (SG14)



## • 滝

- Interactive Procedural Modeling of Coherent Waterfall Scenes (EG15)

# Deep Cherry Tomato



# クイズ: この画像はチェリートマトのどこでしょう

- 1) 花 or 実
- 2) 茎
- 3) 葉っぱ



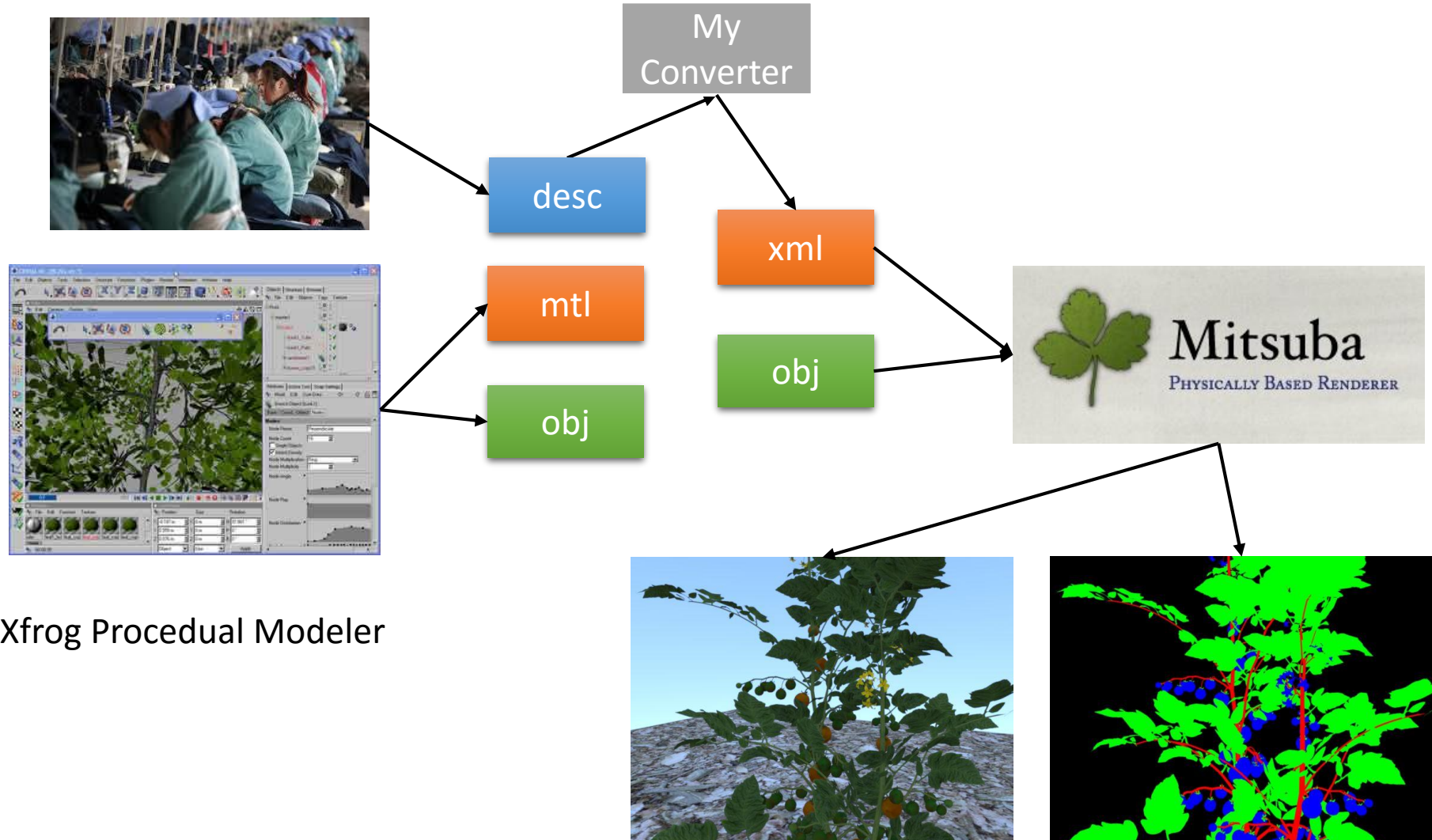
# クイズ: この画像はチェリートマトのどこでしょう

- 1) 花 or 実
- 2) 茎
- 3) 葉っぱ

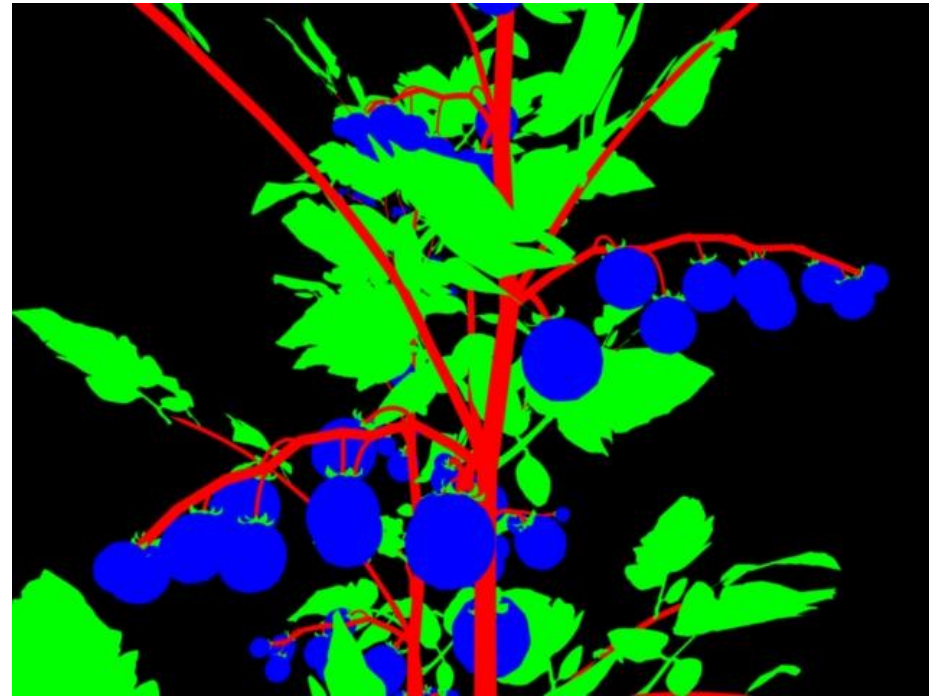


なんか、出来そうな気がする。。。。

# Deep Cherry Tomato Framework



# Deep Cherry Tomato Framework



Rendered  
with





# Deep Cherry Tomato Framework



Rendered  
with



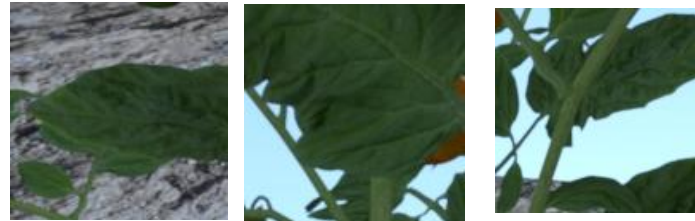
# Deep Cherry Tomato Framework

128x128 の画像 x 50000 x 4 クラス  
(7 モデル x Cherry Tomato から)

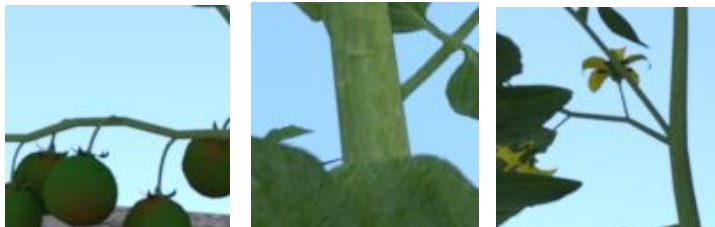
花



葉



茎



Based on tutorials by the Caffe creators at UC Berkeley

## Caffe: Open Source Deep Learning Library



Maximally accurate	Maximally specific
espresso	0.99990
coffee	0.99914
beverage	0.99914
liquid	0.99900
fluid	0.99910

Tutorials by Evan Shelhamer, Jon Long, Sergio Guadarrama, Jeff Donahue, and Ross Girshick and Yangqing Jia



caffemodel

deploy

# Deep Cherry Tomato Framework

Google のイメージ検索の画像を突っ込んで、  
per pixel でセグメンテーション

→Deep Learning の精度をあげるには、

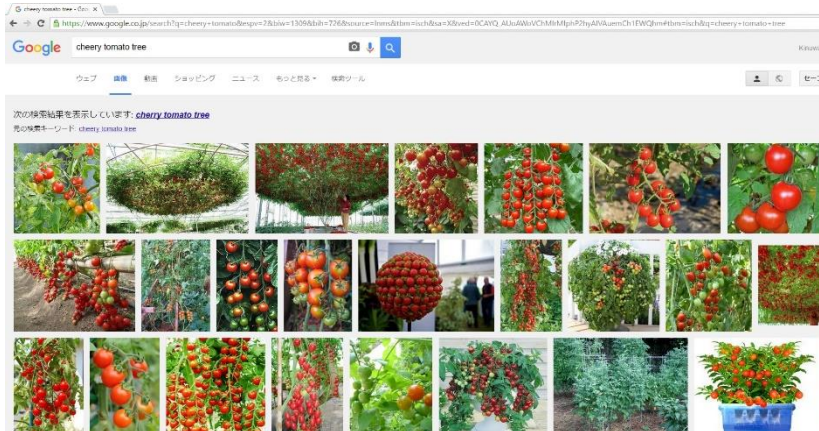
本当は画角とかを合わせて撮影したほうがいいのだが。。。

→手持ちのデジカメの公開情報が少ないため

→チェリートマトってどこで撮影出来るの。。。？

deploy

caffemodel



Based on tutorials by the Caffe creators at UC Berkeley  
Caffe: Open Source Deep Learning Library

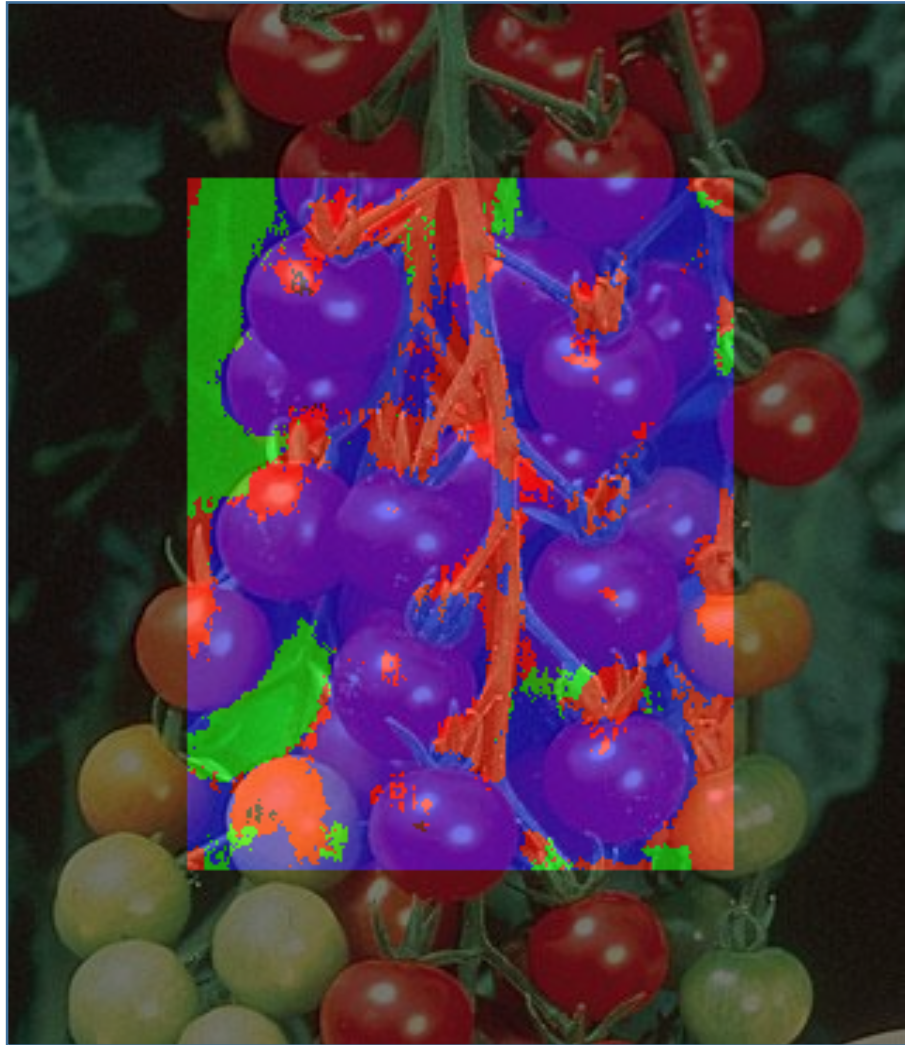


Tutorials by Evan Shelhamer, Jon Long,  
Sergio Guadarrama, Jeff Donahue, and Ross  
Girshick and Yangqing Jia



# 結果





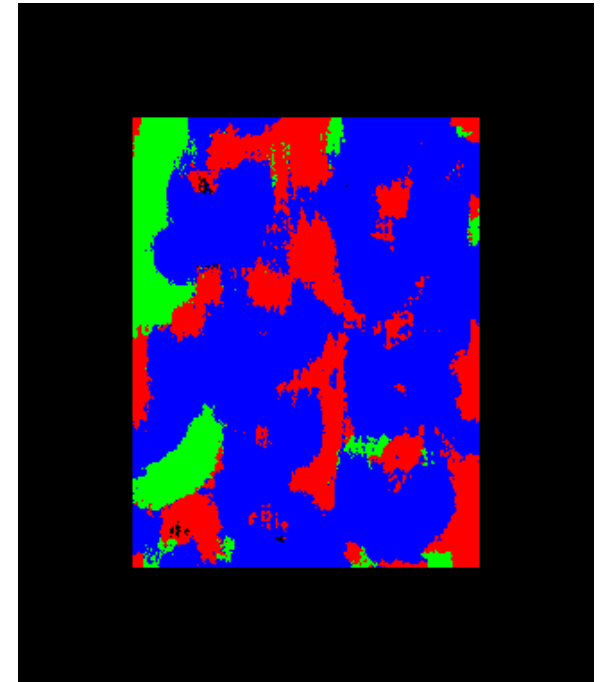
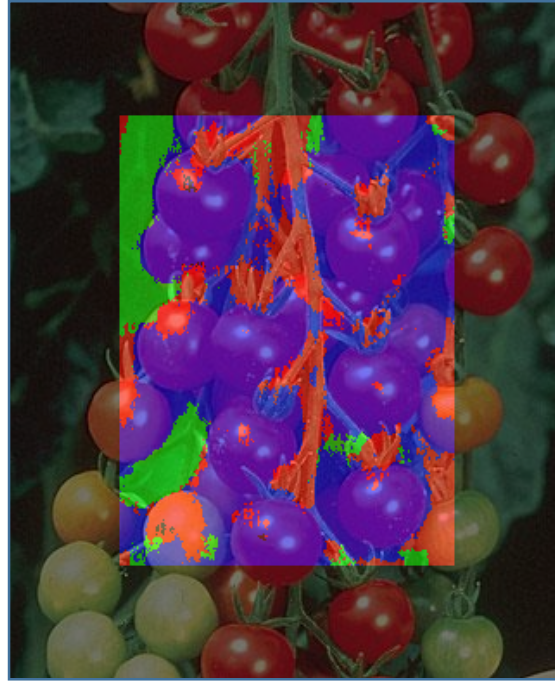
青色:実 or 花  
緑色:葉っぱ  
赤色:茎

トマト多め？

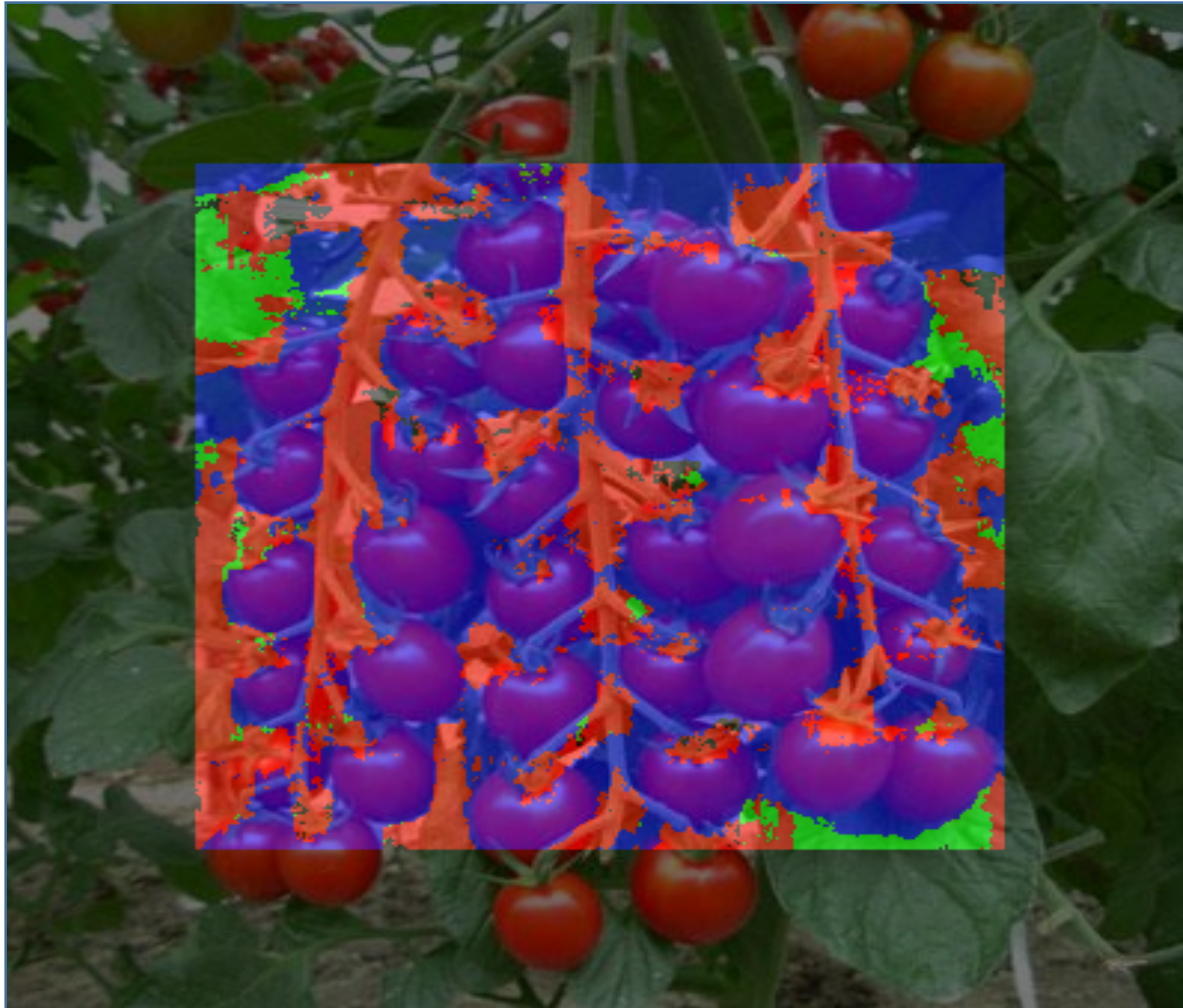
→全体的にざっくりセグメンテーション

→赤色見つけたら問答無用でトマトにする傾向？

→学習にはまったくくないケースだが頑張ってる



青色:実 or 花  
緑色:葉っぱ  
赤色:茎



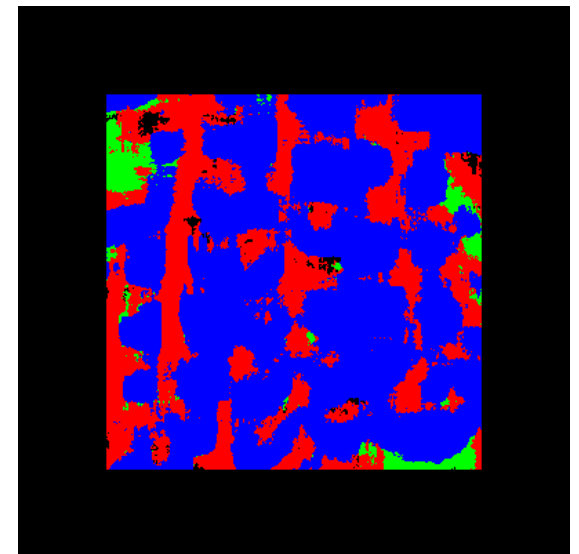
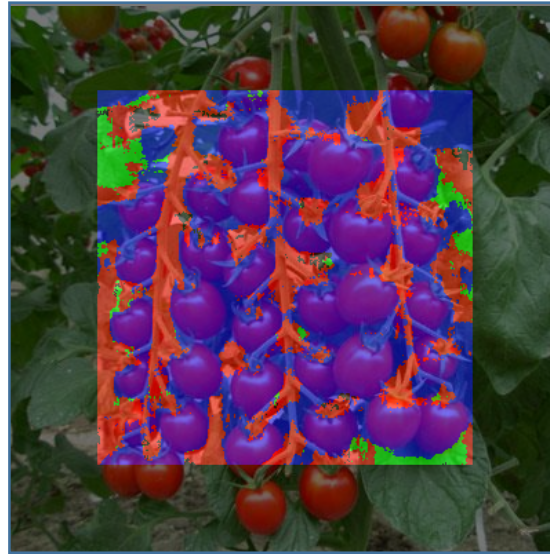
青色:実 or 花  
緑色:葉っぱ  
赤色:茎



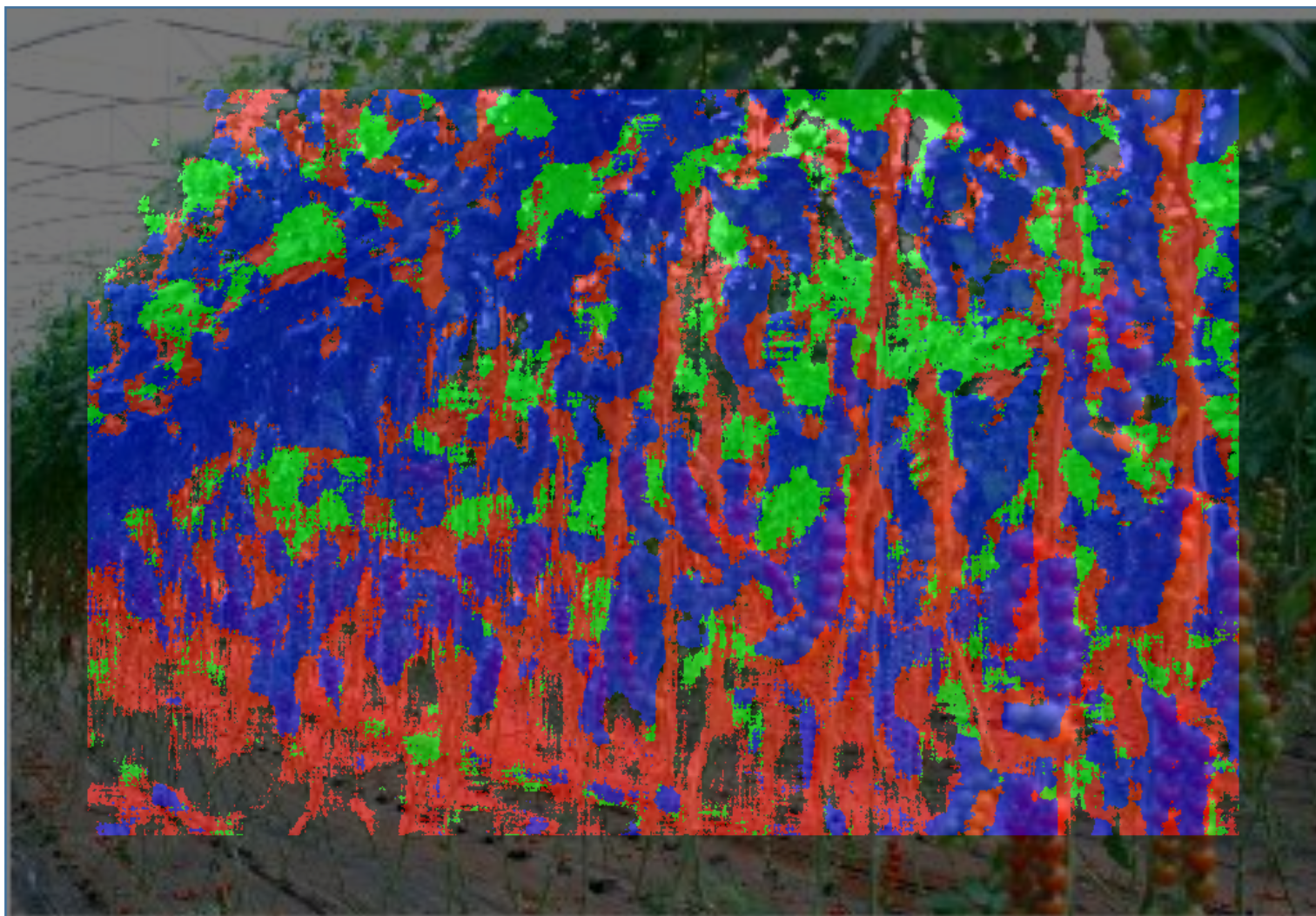
# 描画と実際のシーンの違い

→ トマトにスペキュラが乗ってる

→ トマトが一行に並んでいる(人工的だから?)



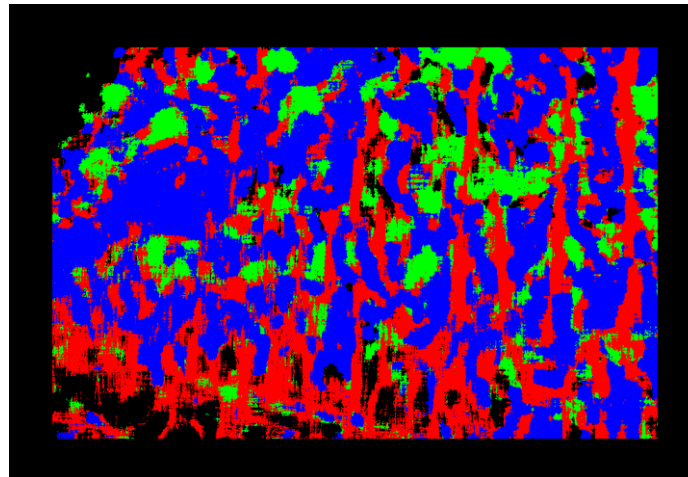
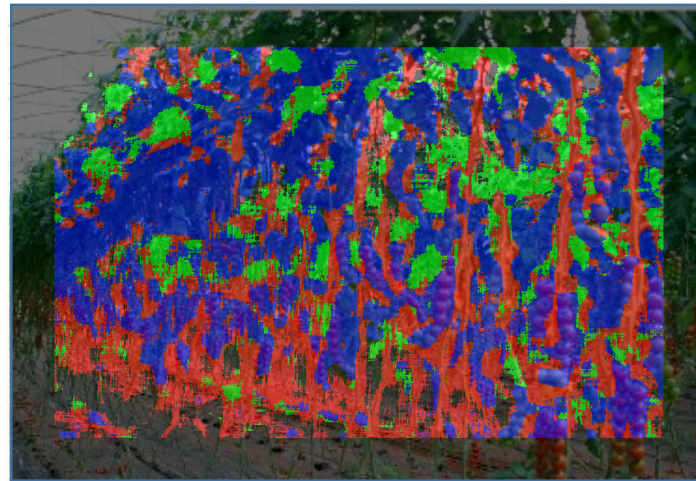
青色: 実 or 花  
緑色: 葉っぱ  
赤色: 茎



青色:実 or 花  
緑色:葉っぱ  
赤色:茎



# 当然だが、サンプルには全くないケース



青色:実 or 花  
緑色:葉っぱ  
赤色:茎

突っ込みどころも一杯  
言いたいことも一杯

# 今日話をしたいこと

	今回	今後に向けて
モデラ (Xfrog)	無料ダウンロード出来る、 7 モデルを使用	自然界にある物が生成出来ない CUI やバッチでモデルが生成出来ない
テクスチャ	無料ダウンロード出来る サンプルを使用	BSDF が無い キャプチャも実測ベースでやるべきかも
天体モデル	Mitsuba の天体モデルを 使用(ランダム)	夕日が出ない。雲も出ない IBL を使った方がいいかもしれない。。。
カメラモデル	Mitsuba で画角をランダムに 変更	手持ちのカメラに合わせた方がよさそう
トーンマップ	Mitsuba の Reihhard を 使用	手持ちのカメラで実測する必要がある？
Deep Learning	Caffe の AlexNet を使用	素人なので勉強しないと
応用例	チェリートマトのみ	他の花、木や街にも応用できるかも？ けど需要ある？