

VRを活用した バーチャルミュージアムの試作

谷研究室 荻山亮大・坂田拓哉

目次

1.はじめに

- 1.1 バーチャルミュージアムとは
- 1.2 バーチャルミュージアムの
利点と欠点
- 1.3 バーチャルリアリティとは
- 1.4 VR化する利点
- 1.5 演習の内容

2.画像の3次元化

- 2.1 コンピュータビジョン
- 2.2 Structure From Motion
- 2.3 Multi View Stereo

- 2.4 使用ソフトウェア

- 2.5 復元

3.アプリケーション開発

- 3.1 開発環境
- 3.2 開発
- 3.3 デモ映像

4.今後の課題

5.参考文献

1.はじめに

1.1 バーチャルミュージアムとは

バーチャルミュージアム

デジタル情報の形でモノを格納し、
ネットワークを通して公開するミュージアム

実在する必要はなく、
その本質を捉えているもの

1.はじめに

1.2 バーチャルミュージアムの利点と欠点

利点

- ・ 時間に制約がない
- ・ いつでも容易に情報収集が可能
- ・ 場所を要さない
- ・ 実物を展示しないため、劣化・損傷を防ぐ
- ・ 展示物の入れ替えが容易

欠点

- ・ 一位の方向からのみの鑑賞
- ・ 現時点,実物とのクオリティの差

1.はじめに

1.3 VR(バーチャルリアリティ)とは

VR ・ ・ ・ Virtual Reality
事実上の 現実

現実には存在していないが、現実の**本質**を有するもの

例)バーチャルマネー(クレジットカード)
実在はしないがお金の**本質**を有するもの

(参考文献:バーチャルリアリティ学)

1.はじめに

1.4 VR化する利点

没入感

- ・ 立体的に捉えることで多角的な鑑賞が可能
- ・ 本来の動きと近い動きで鑑賞することが可能

1.はじめに

1.5 演習の内容

VRを活用したバーチャルミュージアムの試作

- ・ 本学部の所持する土器を撮影し3D化
- ・ PC上に博物館のVR空間を作成
- ・ 3DモデルをVR博物館に置くことで
バーチャルミュージアムを作成

目次

1.はじめに

- 1.1 バーチャルミュージアムとは
- 1.2 バーチャルミュージアムの
利点と欠点
- 1.3 バーチャルリアリティとは
- 1.4 VR化する利点
- 1.5 演習の内容

2.画像の3次元化

- 2.1 コンピュータビジョン
- 2.2 Structure From Motion
- 2.3 Multi View Stereo

- 2.4 使用ソフトウェア

- 2.5 復元

3.アプリケーション開発

- 3.1 開発環境
- 3.2 開発
- 3.3 デモ映像

4.今後の課題

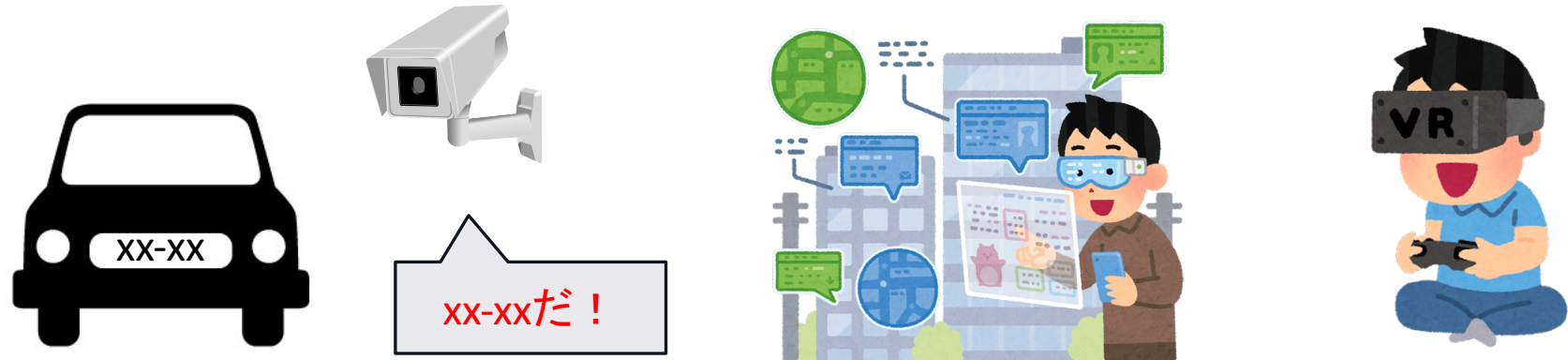
5.参考文献

2.画像の3次元化

2.1 コンピュータビジョン

コンピュータビジョン

人間の視覚的認知をコンピュータで行おうとする分野
2次元画像から3次元空間の復元から始まり、
顔検出や車両ナンバー認識システム,AR,VRなどを指す



2.画像の3次元化

2.2 Structure From Motion

Structure From Motion

動画画像や画像群から画像間の特徴を結びつけることによってカメラの位置・姿勢,そのシーンの特徴点の3次元位置を復元する手法

- ・ 複数枚の画像から疎な点群を取得可能
- ・ 地球科学や文化遺産の構造解析などにも利用



画像引用元:
<https://www.cs.cornell.edu/~snavely/bundler/>

2.画像の3次元化

2.2 Structure From Motion

アルゴリズムの概要

1～3を繰り返す

1. 2つの画像から同一な特定の点を対応付け
2. この対応点からカメラの位置・姿勢情報を取得
3. 対応点とカメラの情報から3次元形状を復元

生じた誤差をこれまで得られた情報からの推定値と実測値の差が最小となる最適解を求め精度を上げる

2.画像の3次元化

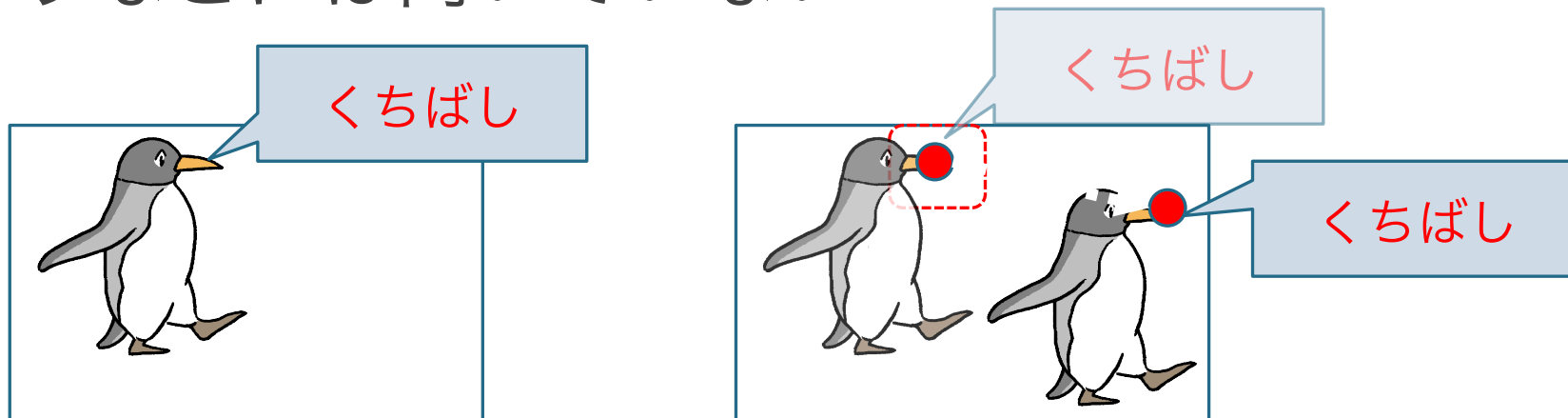
2.2 Structure From Motion

SIFT

SIFTは拡大縮小・回転・明るさの変化に対応出来る

→複数の画像間での特徴点追跡に向いている

一方で計算量が多いためリアルタイム性を必要としているアプリなどには向いていない

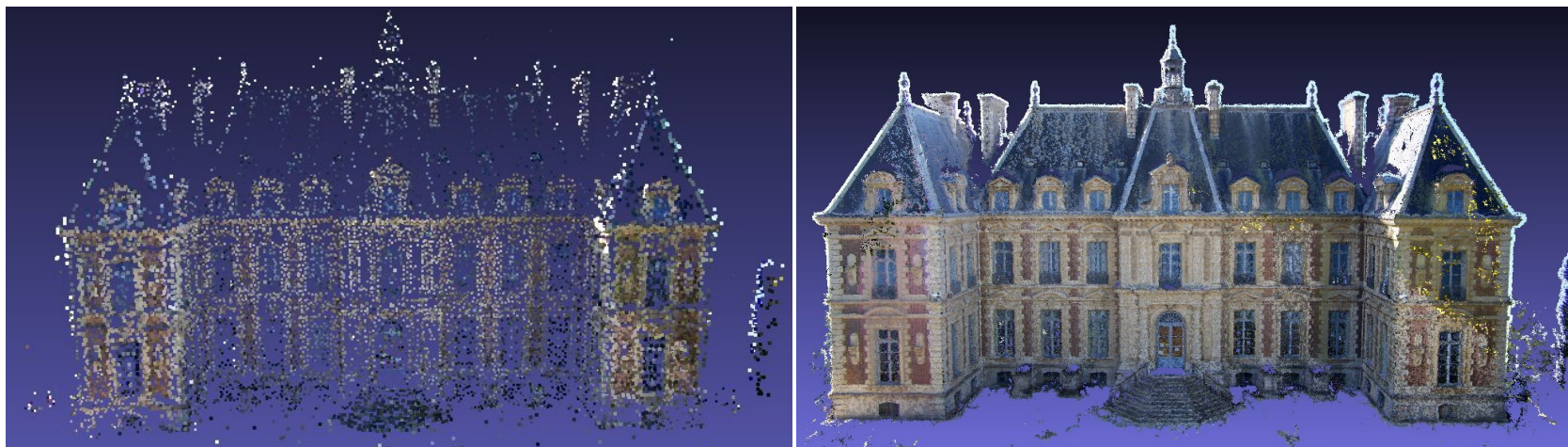


2.画像の3次元化

2.3 Multi View Stereo

Multi View Stereo

Sfmによって得られた3次元点群と
カメラの位置・姿勢情報を用いて
より密な点群を推定し求める手法



画像引用元 : <https://blog.negativemind.com/2016/08/31/openmvs-sample/>

2.画像の3次元化

2.4 使用ソフトウェア

VisualSfm

Structure From Motionを用いた
3次元空間復元のGUIアプリケーション
->点群の生成

Meshlab

3Dのメッシュを処理編集する
オープンソースソフトウェア
->面の生成

2.画像の3次元化

2.5 復元

撮影

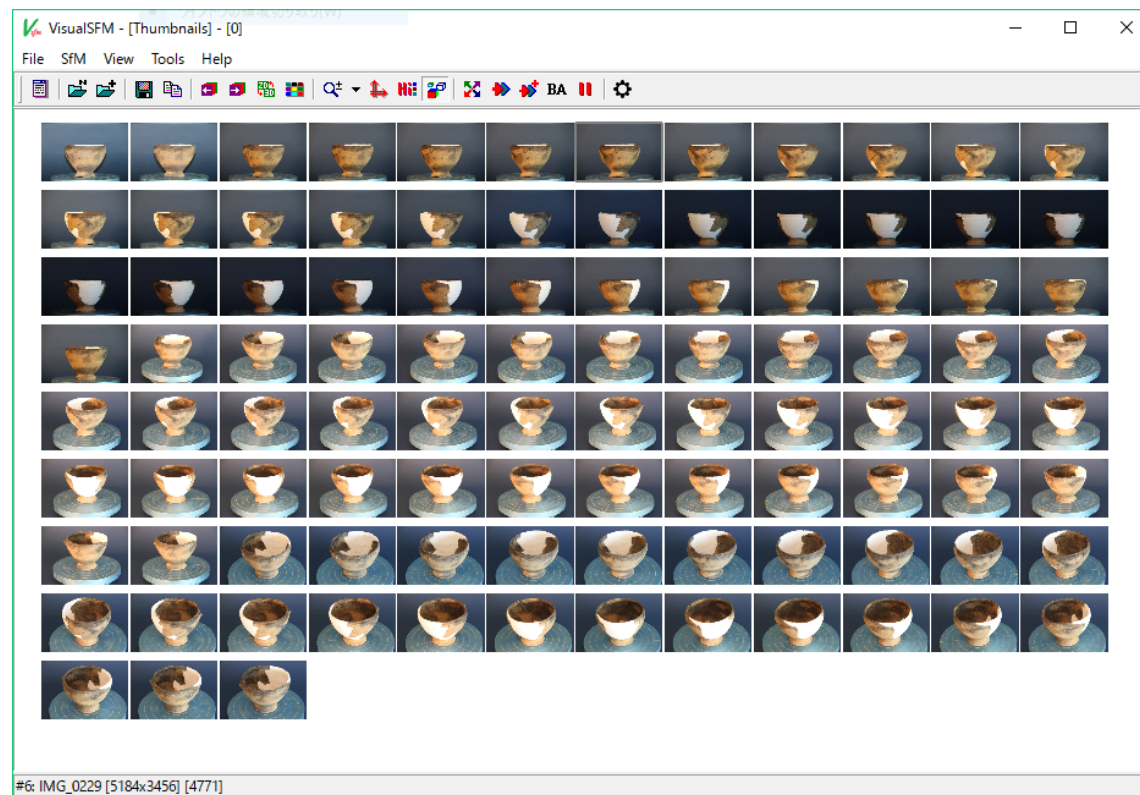


2.画像の3次元化

2.5 復元

visualSfm : 1

- 画像群を読み取る
- 特徴点を抽出

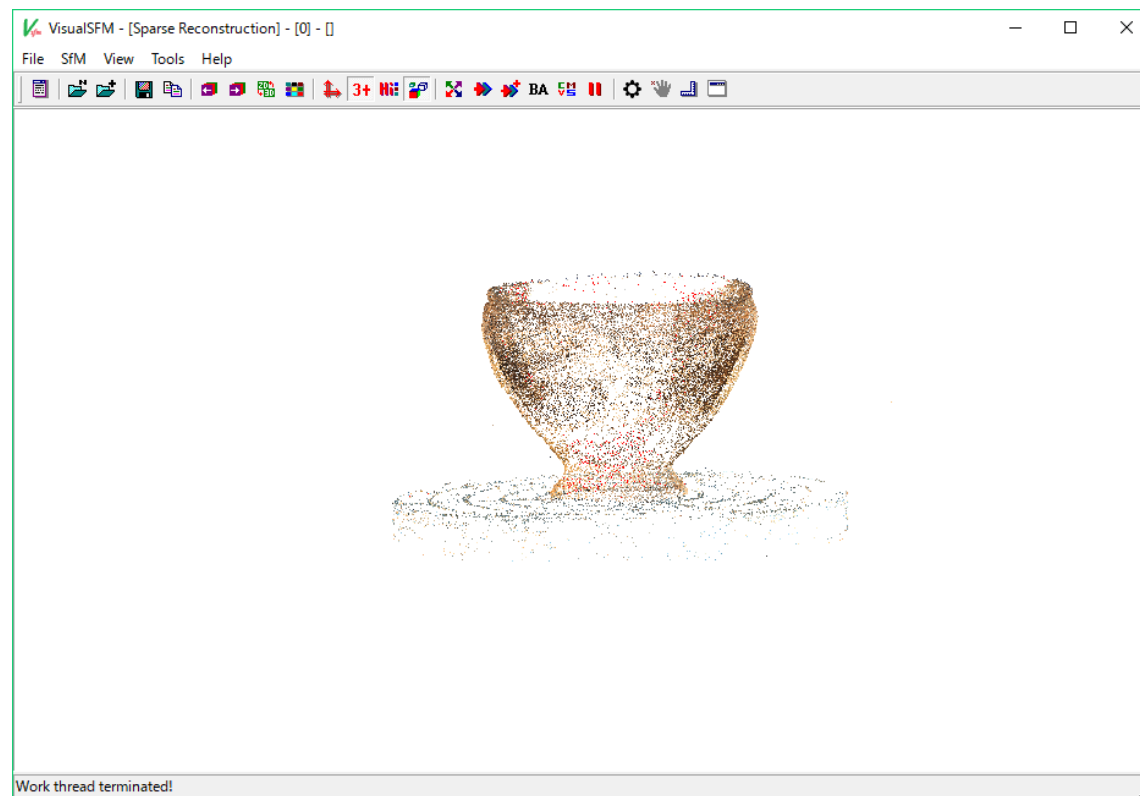


2.画像の3次元化

2.5 復元

visualSfm : 2

- 特徴点を比較
- 類似特徴点を検出
- 3次元空間上に
疎な点群を生成

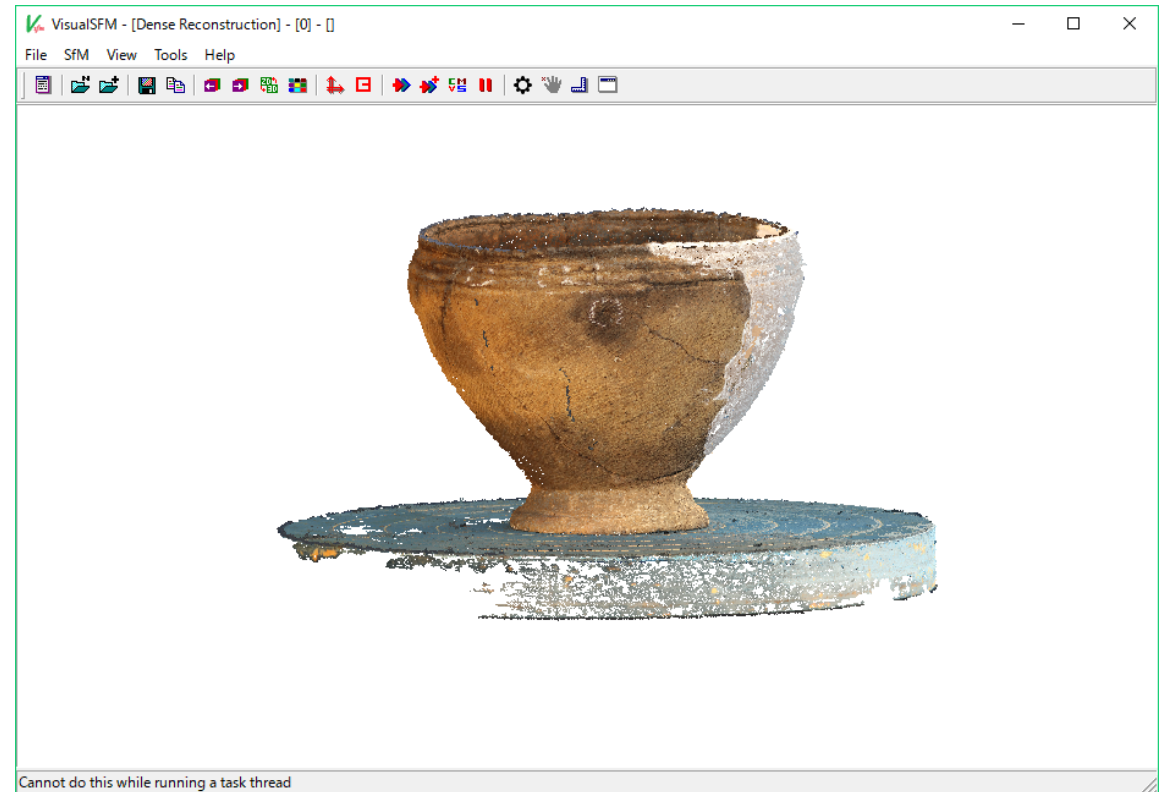


2.画像の3次元化

2.5 復元

visualSfm : 3

- 3次元の疎な点群から予測
- 密な点群を生成

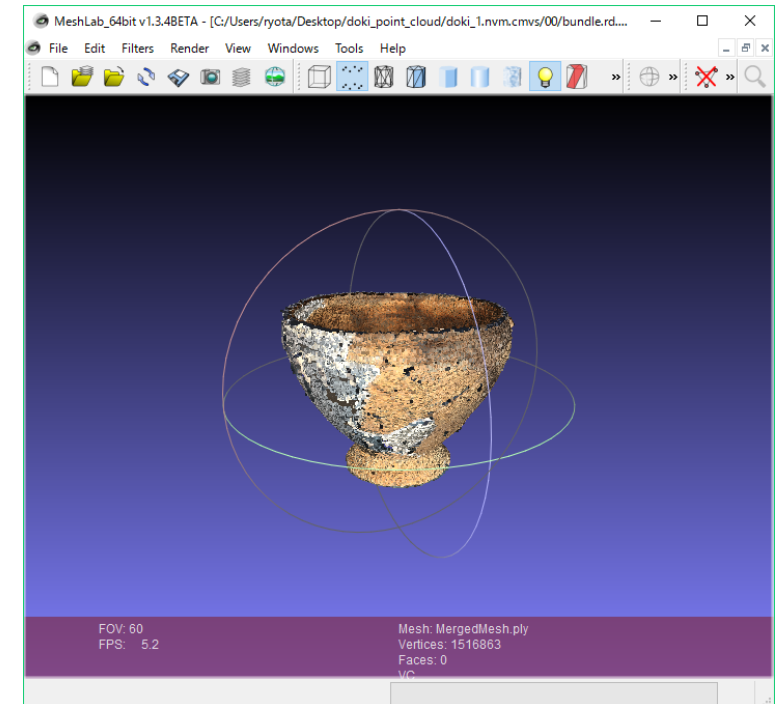


2.画像の3次元化

2.5 復元

Meshlab : 1

- Visualsfmによって得られた
密な点群データを読み込む

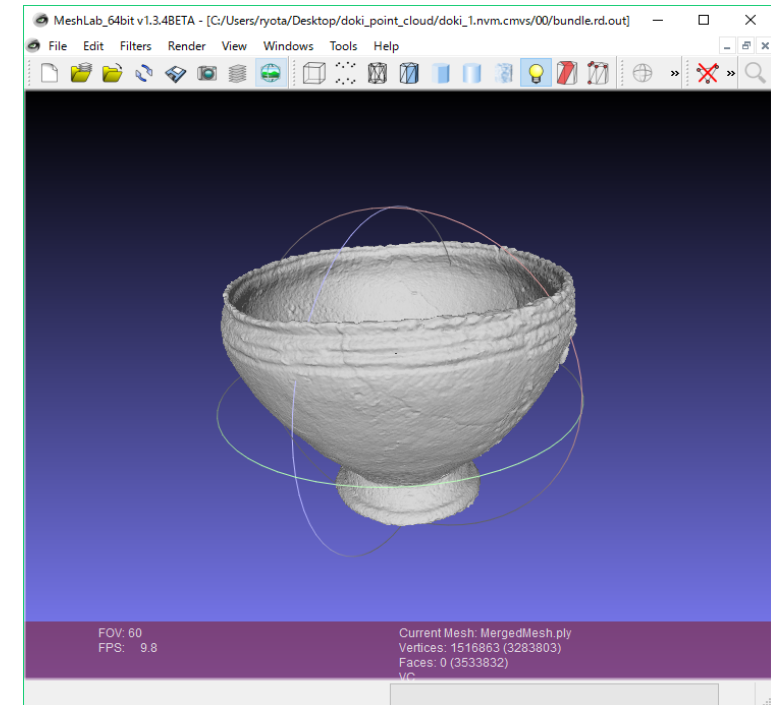


2.画像の3次元化

2.5 復元

Meshlab : 2

- ・ 法線ベクトルを求めメッシュ化
->光の反射や視覚効果を再現

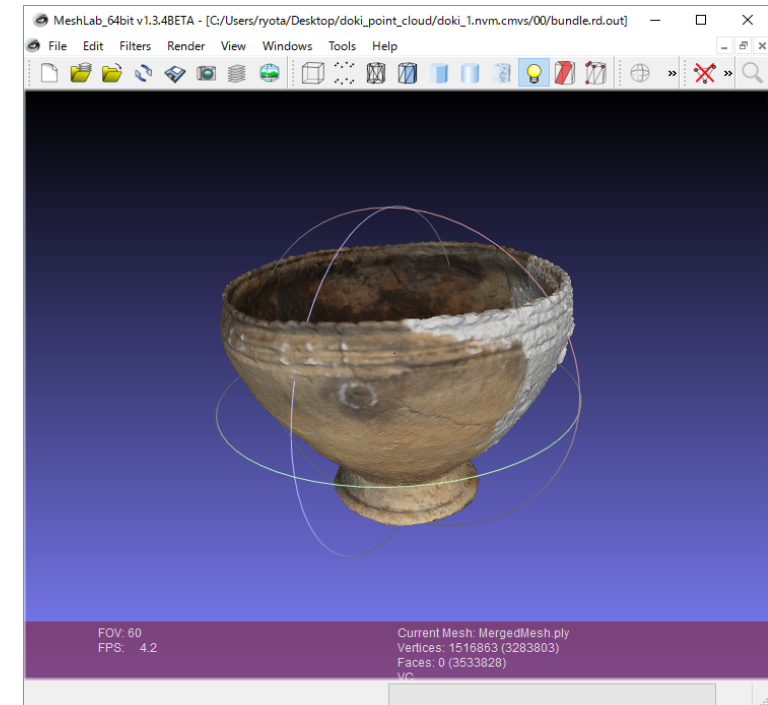


2.画像の3次元化

2.5 復元

Meshlab : 3

- ・メッシュにテクスチャを貼る



目次

1.はじめに

- 1.1 バーチャルミュージアムとは
- 1.2 バーチャルミュージアムの利点と欠点
- 1.3 バーチャルリアリティとは
- 1.4 VR化する利点
- 1.5 演習の内容

2.画像の3次元化

- 2.1 コンピュータビジョン
- 2.2 Structure From Motion
- 2.3 Multi View Stereo

- 2.4 使用ソフトウェア

- 2.5 復元

3.アプリケーション開発

- 3.1 開発環境
- 3.2 開発
- 3.3 デモ映像

4.今後の課題

5.参考文献

3.アプリケーション開発

3.1 開発環境

使用ソフトウェア

Unity

WindowsとOS X上で動作する統合型ゲーム開発環境

MonoDevelop

Unityの提供する統合開発環境

使用言語はC#

3.アプリケーション開発

3.1 開発環境

使用ハードウェア

Oculus Rift CV1

Oculus社が開発したバーチャルリアリティ向け
ヘッドマウントディスプレイ



画像引用元：

<https://www3.oculus.com/en-us/rift/>

3.アプリケーション開発

3.2 開発

自分自身(カメラ)の移動はオキュラスリモートを使用



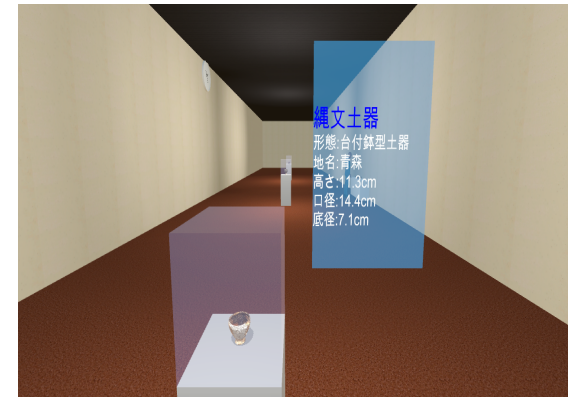
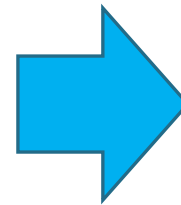
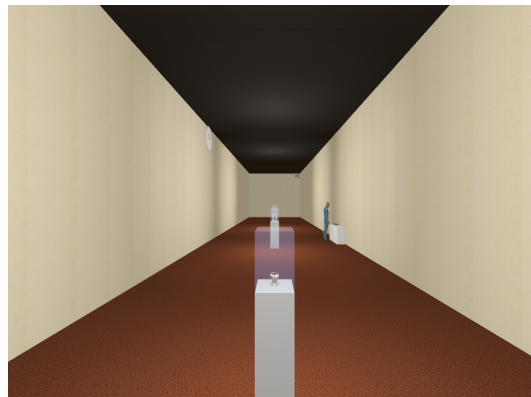
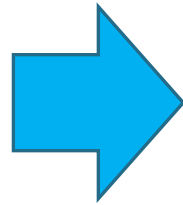
画像引用元:

<http://www.itmedia.co.jp/news/articles/1601/07/news053.html>

3.アプリケーション開発

3.2 開発

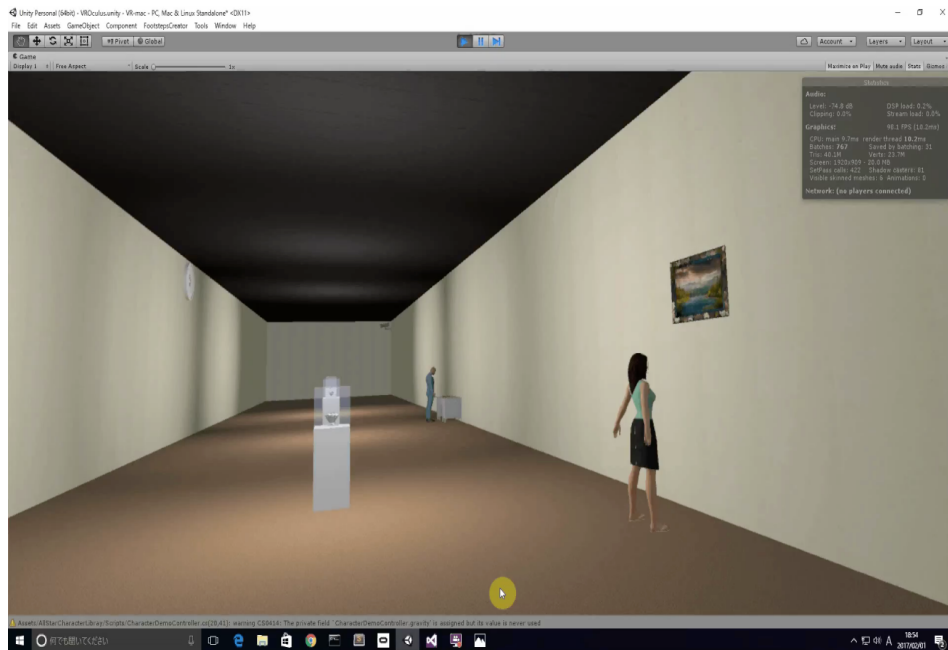
バーチャルミュージアムの流れ



3.アプリケーション開発

3.3 デモ映像

ディスプレイの動画



周りから見た動画



目次

1.はじめに

- 1.1 バーチャルミュージアムとは
- 1.2 バーチャルミュージアムの
利点と欠点
- 1.3 バーチャルリアリティとは
- 1.4 VR化する利点
- 1.5 演習の内容

2.画像の3次元化

- 2.1 コンピュータビジョン
- 2.2 Structure From Motion
- 2.3 Multi View Stereo

- 2.4 使用ソフトウェア

- 2.5 復元

3.アプリケーション開発

- 3.1 開発環境
- 3.2 開発
- 3.3 デモ映像

4.今後の課題

5.参考文献

4. 今後の課題

- ・ 展示物の種類を増やす
- ・ 画像を入れるだけで,自動的に配置されるUIの作成
- ・ 操作の簡易化
- ・ 被験者を募り,アプリケーションの利便性の向上

目次

1.はじめに

- 1.1 バーチャルミュージアムとは
- 1.2 バーチャルミュージアムの
利点と欠点
- 1.3 バーチャルリアリティとは
- 1.4 VR化する利点
- 1.5 演習の内容

2.画像の3次元化

- 2.1 コンピュータビジョン
- 2.2 Structure From Motion
- 2.3 Multi View Stereo

- 2.4 使用ソフトウェア

- 2.5 復元

3.アプリケーション開発

- 3.1 開発環境
- 3.2 開発
- 3.3 デモ映像

4.今後の課題

5.参考文献

参考文献

- http://www.civil.chuo-u.ac.jp/lab/keisan/VR_mini_museum/newpage2.html
- <http://www.forum8.co.jp/topic/3Dcontents102.htm>
- <http://mcm-www.jwu.ac.jp/~physm/buturi06/sikaku/st1.html>
- 点対応を用いた複数の2次元画像からの2次元形状復元-因子分解法の数理-
産業技術総合研究所 藤木 淳
- 画像局所特徴による対応点マッチングと その応用に関する研究

情報工学専攻

氏名 西村 孝

PCスペック

	推奨PC	使用PC
OS	Windows7 SP1 64bit以上	Windows 10 Pro 64bit
CPU	Intel core i5 4590以上	Intel Core i7 6700(3.4GHz)
メモリ	8GB以上	16GB
GPU	NVIDIA Geforce GTX 970 以上	NVIDIA GeForce GTX 1070 (1506MHz)
USB	3.0ポート x 3 2.0ポート x 1	3.0ポート x 4 2.0ポート x 4

VR酔い

VR酔い

自分が体感・予測している感覚,
記憶している感覚との間にずれ

VR酔いの対策

- ・ 歩行の際の急激な加減速の控え
- ・ 頭の向きと一致したカメラの向き



立体視の原理

人が立体を感じる心理的原理

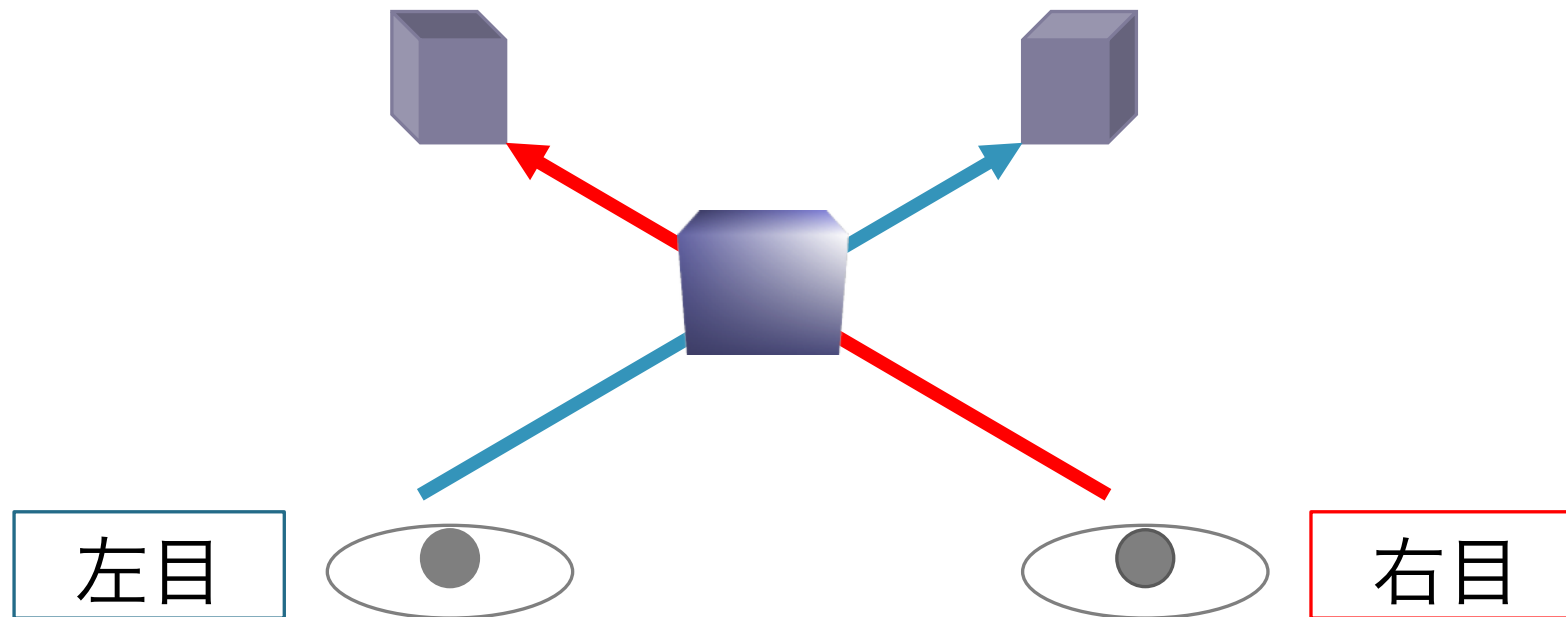
- ・ 遠くのものは小さく見える
- ・ 手前のものは奥のものを隠す
- ・ 遠くのほうが霞んで見える
- ・ 陰影
- ・ 手前のほうが早く動く(ように見える)



立体視の原理

人が立体を感じる**生理的原理**

両眼視差 左右の目の網膜像の違い



立体視の原理

輻輳	対象物を注視するときの眼球の動き(寄り目)
運動視差	観察者と対象物の相対的な運動によって生じる 網膜像の変化
焦点調節	水晶体の調節によるピント合わせ (VRデバイスでは投影面が固定されているため 行えない)

VRデバイスでは両眼視差、輻輳、運動視差で立体視を生成