Development and Application of a Description-based Interface for 3D Object Reconstruction

Kai Wu

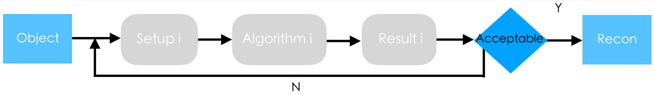
October 20, 2017

University of British Columbia
kaywu@ece.ubc.ca

Table of contents

Motivation

Motivation: traditional 3D Reconstruction

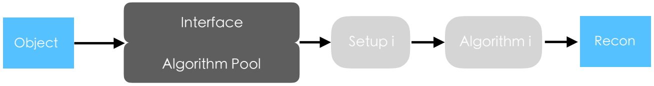


Challenges:

- Hardware: controlled environment, calibration
- Algorithms: vision background
- Results: keep trying until an acceptable result

Motivation: interface to 3D Reconstruction

What if we can create an interface above the 3D algorithms, which can select an appropriate algorithm based on users' description, and achieve a successful reconstruction result.



Contribution

Contribution

Development of an interface for 3D reconstruction problem, which hides algorithmic details and allows users to describe conditions surrounding the problem. This description can be interpreteded so that an appropriate algorithm is chosen to obtain a successful reconstruction result.

Contribution (cont'd)

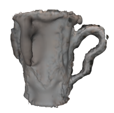
This contribution is significant because:

- No single algorithm can work for a diverse categories of objects. The interface, to some extent, can cover a wider range of object categories by incorporating multiple algorithms.
- An description is provides that hides the algorithmic details, thus understanding of the algorithm, or conditions to apply a specific algorithm is not a prerequisite.

Related Work

Related Work: softwares

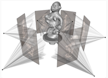
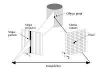
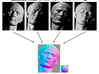
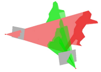
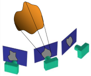
Some noteable examples of open source or commercial softwares:

Case	Software	Example	Problem
General VL	OpenCV	Mat findHomography();	Generic APIs
Software bundles	SfM + MVS	 	Cater to specific objects

Challenge

Not that we don't have enough tools, but the barrier to take advantage of these tools is high.

Related Work: algorithms

Category	Method	Problem
Shape from Stereo		Texture, Specular
		Albedo, Specular
Shape from Intensity		Albedo, Specular, Geomtry
		Albedo, Geoemtry
Shape from Silhouette		Geoemtry
		Geoemtry

Related Work: algorithms

Challenges

1. No algorithm works for objects with diverse properties;
2. The knowledge of whether an algorithm works is not known a priori.

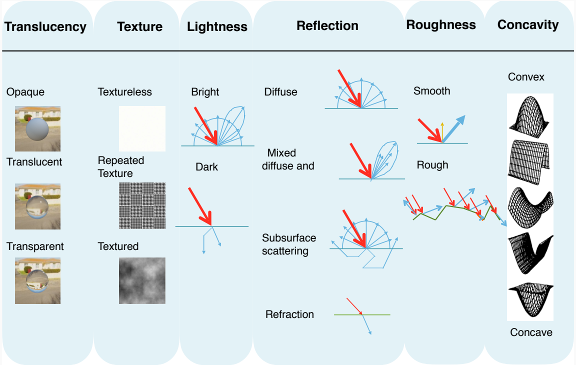
Development of Interface

Roadmap

- Taxonomy: understand the problem space;
- Description: describe the problem space;
- Mapping: discover the connection between a sub-volume in problem space to algorithm.

Taxonomy: properties of problem space

- *algorithm-centered* taxonomy categorizes algorithms based on algorithmic details, as discussed in **Related Work**;
- *object-centered* taxonomy categorizes algorithms based on the problem conditions that the algorithm can reliably works under.



Taxonomy: problem conditions

Assumptions:

- Active methods require high surface albedo (bright), in order to demonstrate the effectiveness of these methods, we focus on bright surfaces only.
- Diffuse is caused solely by surface roughness since sub-surface scattering is ignored.

Condition	Texture		Lightness		Reflection		Roughness		Label
	Textureless (Tl)	Textured (T)	Dark (D)	Bright (B)	Diffuse (D)	Mixed (M)	Smooth (S)	Rough (R)	
1	Yes			Yes	Yes			Yes	Tl-B-D-R
2	Yes			Yes		Yes	Yes		Tl-B-M-S
3		Yes		Yes	Yes			Yes	T-B-D-R
4		Yes		Yes		Yes	Yes		T-B-M-S

Description: model and representations

Model	**Representation**
Texture	*Texture randomness*
Lightness	*Albedo*
Specularity	*Specular/diffuse ratio*
Roughness	*SD of facet slopes*

Table 1: Representations of the 3D reconstruction problem.

Description: expression

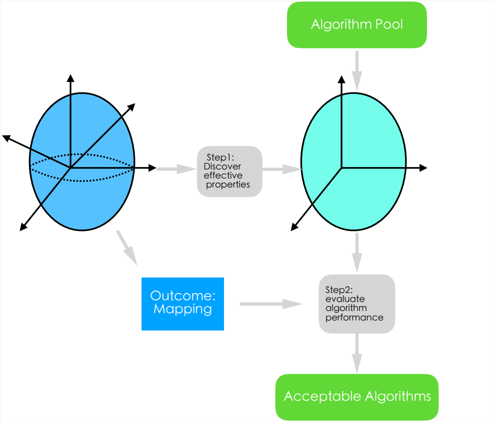
We use three discrete scales to parameterize these properties: *low* (0.2), *medium* (0.5), and *high* (0.8).

Object	Texture	Albedo	Specular	Rough	**Label**
Class 1	low/med	high	low/med	high	Tl-B-D-R
Class 2	low/med	high	high	low/med	Tl-B-M-S
Class 3	high	high	low/med	high	T-B-D-R
Class 4	high	high	high	low/med	T-B-M-S
Class 5	high	low/med	low/med	high	T-D-D-R
Class 6	high	low/med	high	low/med	T-D-M-S

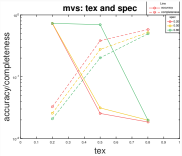
Table 2: Expression of the reconstruction problem for the object classss.

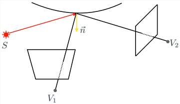
Mapping

Investigate the problem conditions under which the algorithms can reliably work. This strucure of this chapter is as follows

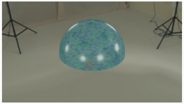


Mapping: notable findings 1

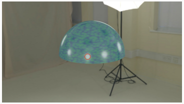




(a). Algo. performance (b) Image formation



(c) V_1

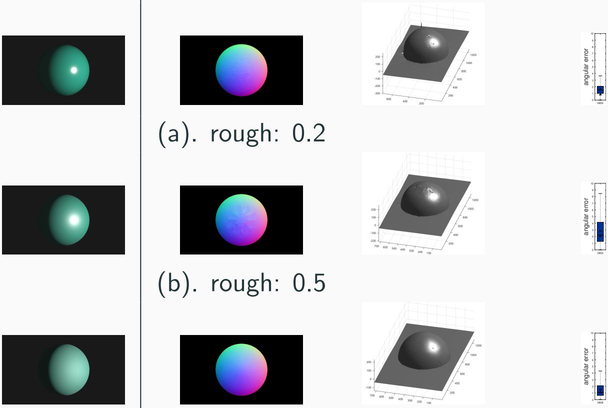


(d) V_2

Figure 1: (a) shows the algorithm performance w.r.t. texture and specularity. (b) shows the reflection of light off a specular surface. V_1 received the diffuse component while V_2 receives the specular component. (c), (d) shows the images observed from these two views. The specular area (red circle) observed in V_2 is visible in V_1.

Mapping: notable findings 2

Image | Normal map | Height map | Angular error



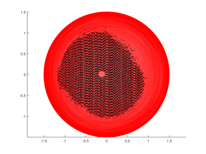
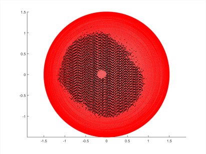
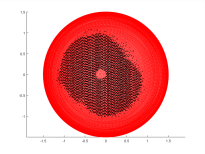
(a). rough: 0.2

(b). rough: 0.5

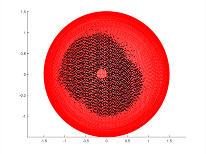
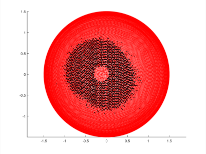
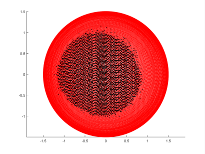
(c). rough: 0.8

Figure 2: The effect of roughness on PS. Albedo is set as 0.8, and specular is set as 0.8. (b) demonstrates that a medium level roughness would lead to worse normal estimation since it blurs the specular lobe.

Mapping: notable findings 3

(a) specular: 0.2 (b) specular: 0.5 (c) specular: 0.8

(d) roughness: 0.2 (e) roughness: 0.5 (f) roughness: 0.8

Figure 3: (a)-(c): the roughness is set as 0.2, and specular has a negative effect on completeness; (d)-(e): the specular is set as 0.8, roughness has a positive effect on completeness.

Conclusions

- PMVS can work on specular surfaces;
- EPS and GSL fails on highly specular areas, and a blurred specular area causes worse results.

Evaluation of Interface

Interpretation: Key Evaluation Questions

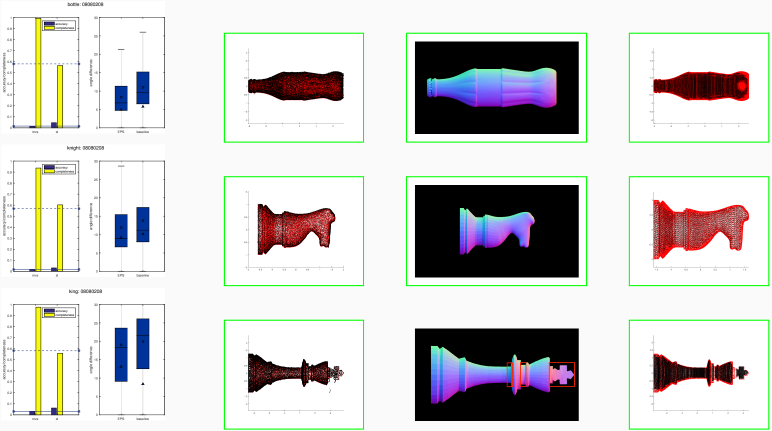
- Evaluation of mapping: is the mapping robust to changes of shape?
- Evaluation of interpreter: can the proof-of-concept interpreter return a successful reconstruction given the correct description?

Interpretation: evaluation of mapping

- Shape variation: too vast and complicated to model;
- Instead focus on one geometric property: surface concavity, and see how robust is the mapping with respect to concavity changes.

Interpretation: evaluation of mapping (cont'd)

Quantitative results	Qualitative results



PMVS EPS GSL

Figure 4: Problem condition: 08080208, mapped algorithms: PMVS, EPS, GSL

Interpretation: evaluation of mapping (cont'd)

Conclusion:

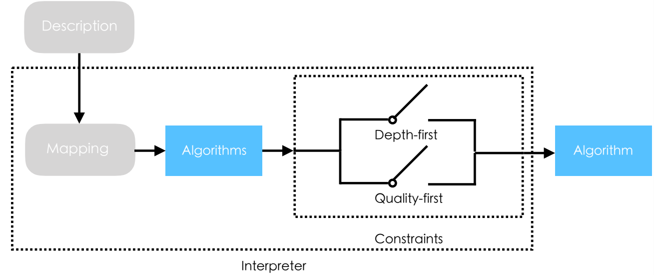
- Mappings to PMVS and GSL are robust to concavity changes whereas those to EPS are not.

Suggestions:

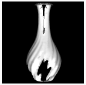
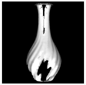
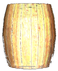
- Develop more advanced description, incorporating concavity into the description
- Use more underlying algorithms that are robust to concavity changes.

Proof-of-concept interpreter

An interpreter selects an appropriate algorithm based on description of problem condition and constraints.



Interpretation: evaluation of interpreter

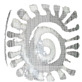
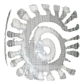
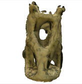
Desc #	Bust	Vase1	Barrel	Vase0	Selected Algo.
1					GSL
2					EPS
3					GSL
4					PMVS

Interpretation: real-world objects

class #	1	2	3	4
description	textureless diffuse bright	textureless mixed d/s bright	textured diffuse dark/bright	textured mixed d/s dark/bright
object				

Figure 5: The rerepsentatives of the four classes of objects used for evaluation.

Interpretation: evaluation of interpreter (cont'd)

Desc #	Statue	Cup	Pot	Vase	Selected Algo.
1					GSL
2					EPS
3					GSL
4					PMVS

Conclusions

Conclusions

- the proposed description is able to give correct reconstruction for non-concave objects
- To deal with more complicated objects, we need more complicated properties, or ways to describe the objects, but the challenge is the easy mathematical representation might not be available.
- Using the simple descriptive language and proof-of-concept interpreter, we demonstrate the possibility of using descriptive properties to hide algorithmic details.

Take-away message

Computer vision should focus on more than just algorithms, but easier accessibility.