Bachelor's Thesis

Towards a real-time 3D object recognition pipeline on GPGPU computing platforms using low-cost RGB-D sensors

June 9, 2015

Alberto García García
< agg180@alu.ua.es >

Supervisors:
José García Rodríguez
Sergio Orts Escolano

Department of Computer Technology (DTIC)
University of Alicante

Table of Contents

Introduction

- Collaboration with the Department of Computer Technology
  - **Research collaboration grant (MECD)**
    - November 2014 - June 2015
  - Research tasks related to
    - Computer vision
    - High Performance Computing
  - **SIRMAVED national project**
    - *Development of a comprehensive robotic system for monitoring and interaction for people with acquired brain damage and dependent people*
    - It requires an onboard 3D object recognition system

Introduction
What is 3D object recognition?

Figure: Scene which contains multiple objects.

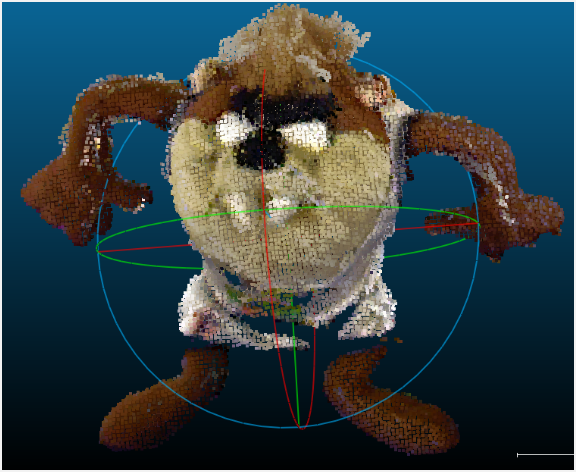


Figure: Taz 3D reconstructed model.

- Object recognition is a very challenging problem
- Traditional approaches based on 2D images are problematic
- 3D based methods open new possibilities in this area
- The advent of low-cost RGB-D sensors shifted the paradigm
- But...
  - High levels of noise and outliers
  - Computationally expensive
- Lack of real-time recognition systems on embedded platforms

- A 3D object recognition system
  - Interactive
    - CPU optimizations
    - GPU parallelization
  - Based on local 3D information
  - Full pose estimation
  - Low-cost 3D sensors
  - Deployed on a embedded platform (Jetson TK1)

Methodology

Methodology

Software

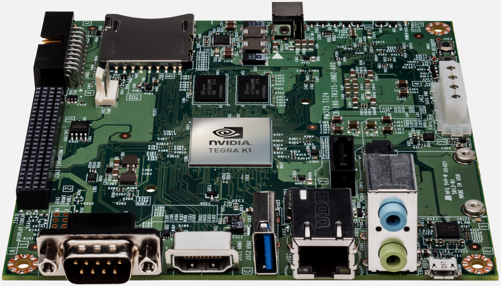
OpenNI™

pointcloudlibrary

OpenMP™

NVIDIA CUDA

XBOX 360



NVIDIA
TEGRA K1

3D object recognition pipeline

3D object recognition pipeline
General pipeline

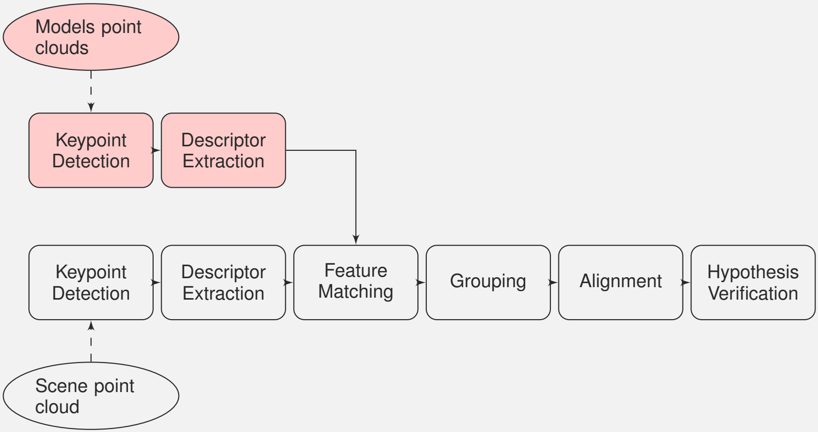


Figure: Object recognition pipeline. Red phases are performed offline while blank ones are live as the sensor provides information.

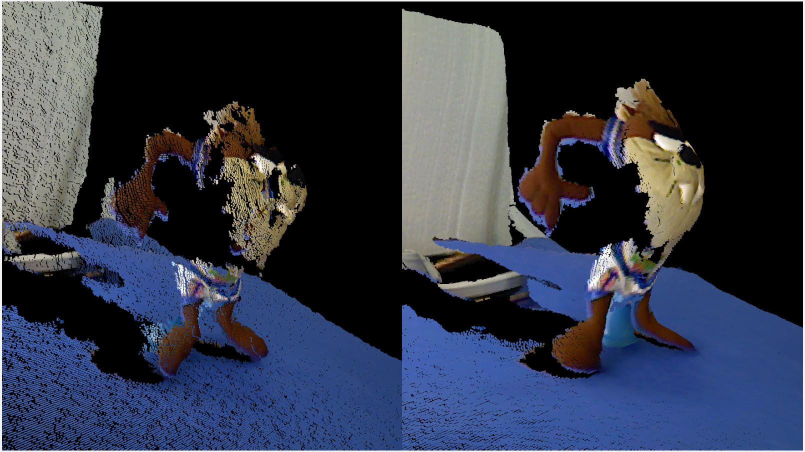


Figure: Comparison of a raw cloud and a bilateral filtered one.

3D object recognition pipeline
Preprocessing: normal estimation

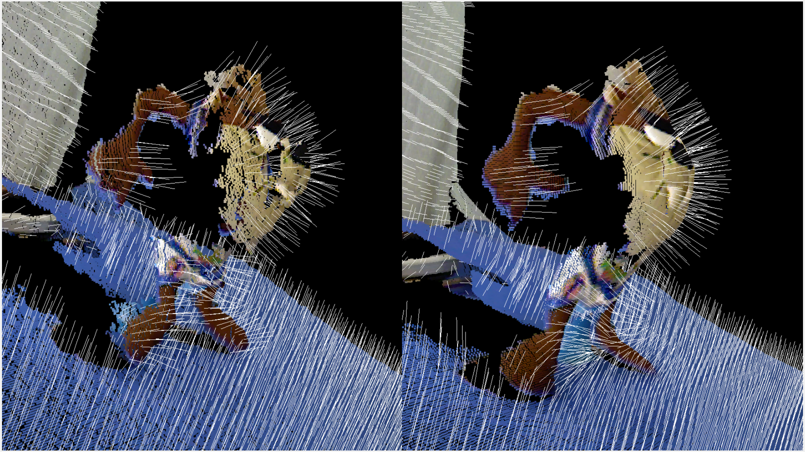


Figure: Comparison of normals of a raw cloud and a bilateral filtered one.

3D object recognition pipeline
Preprocessing: plane segmentation

Figure: Scene captured with PrimeSense Carmine

Figure: Scene segmented with OMPS/OCCS (right) approximately 14% of the original points.

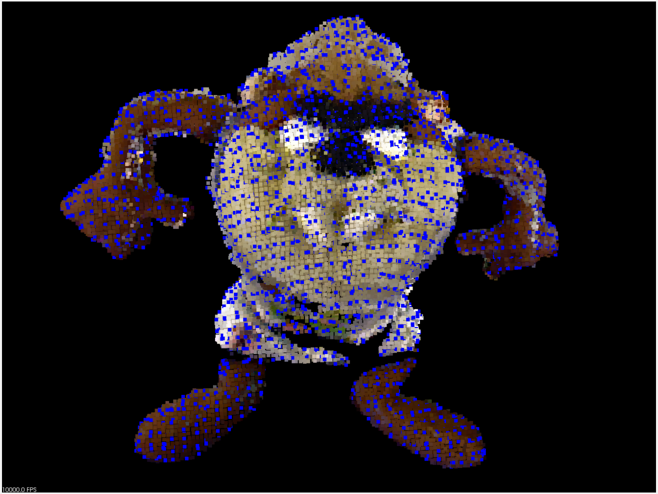


Figure: Keypoints extracted with uniform sampling in blue.

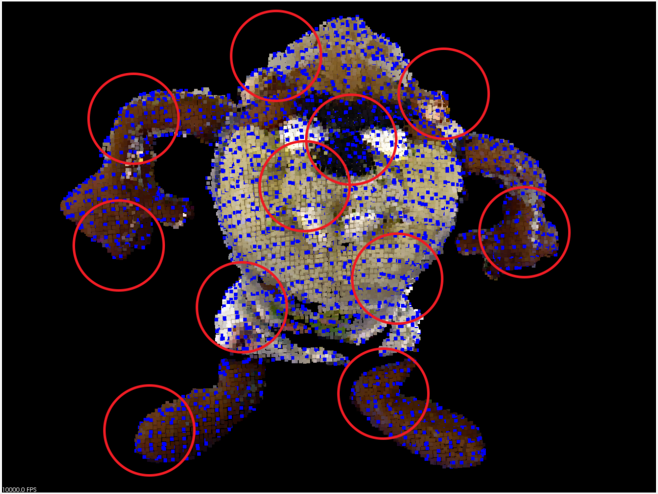


Figure: Descriptor supports shown as red circles.

3D object recognition pipeline
Feature matching and correspondence grouping

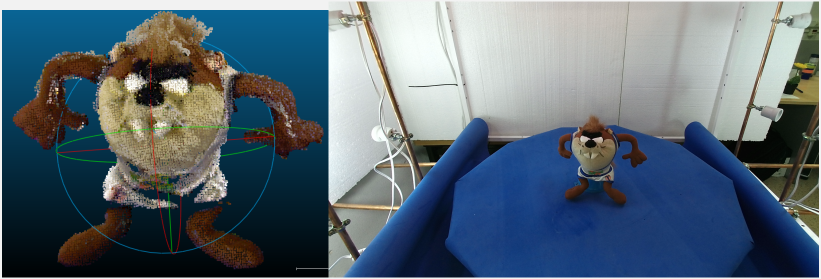


Figure: Model to be recognized (left) and scene (right).

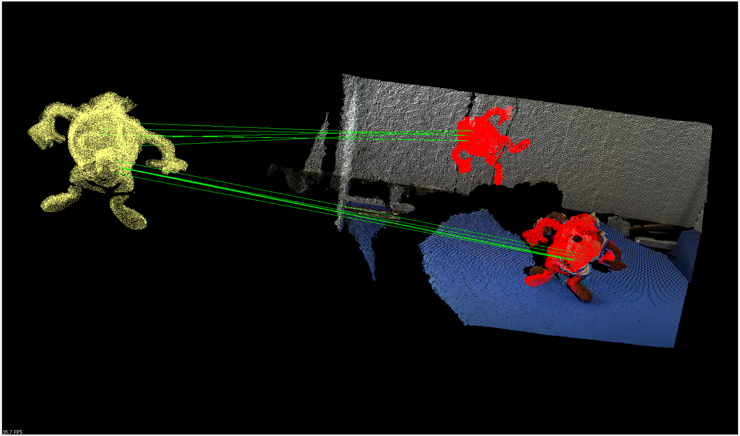


Figure: Model (in yellow), correspondences between model and scene (green) and estimated hypotheses with clustered correspondences (red).

3D object recognition pipeline
Pose refinement

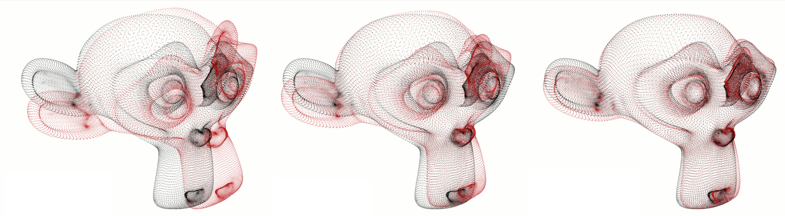


Figure: Cloud alignment refinement with ICP algorithm. The original point cloud in gray and the ICP aligned cloud in red.

3D object recognition pipeline
Hypothesis verification

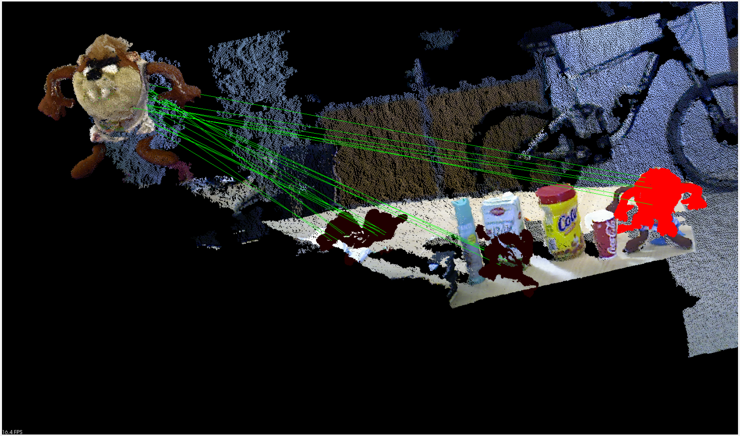


Figure: Model (top left), clustered correspondences (green) with possible hypotheses (red) and the verified true positive hypothesis (light red).

Descriptors and pipeline performance study

Figure: Reconstructed 3D models: Tazmanian (left) and Colacao (right).

Descriptors and pipeline performance study

Methodology (Evaluation scenes)

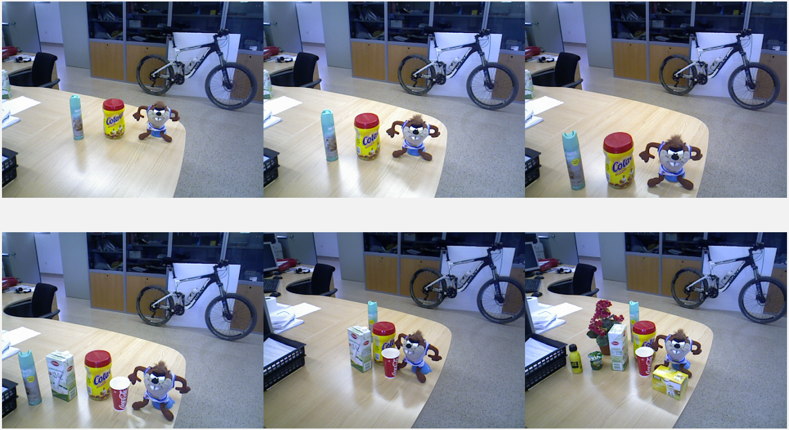


Figure: Heterogeneous test scenes captured with Primesense Carmine.

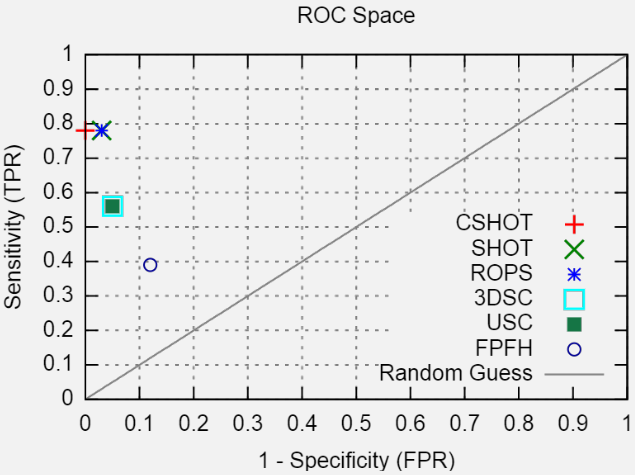


Figure: Descriptor sensitivity and specificity in ROC space.

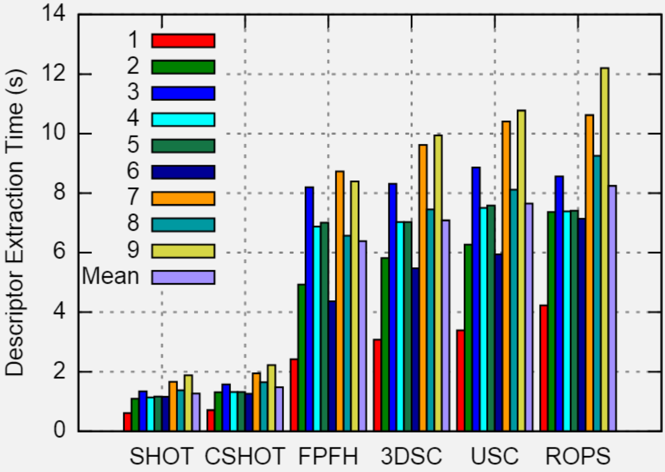


Figure: Descriptor runtimes for the different scenes.

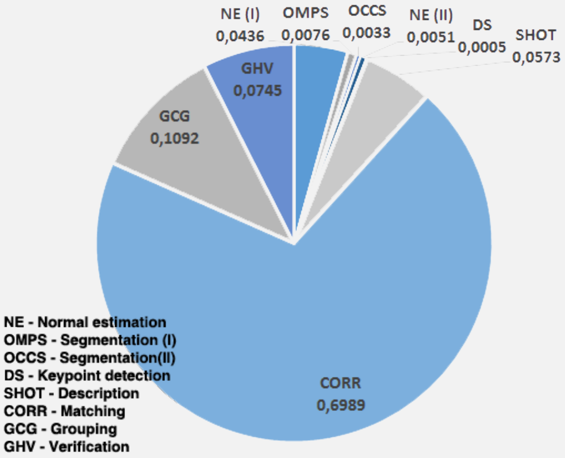
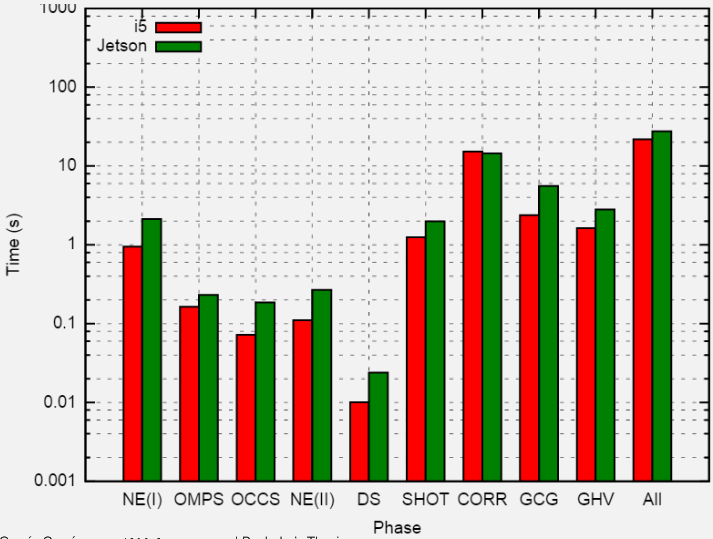


Figure: Pipeline's time distribution.

Descriptors and pipeline performance study

SHOT-based pipeline on Jetson TK1 performance study



- CSHOT is the most accurate descriptor
- SHOT is the fastest descriptor
- Pipeline runtime (with SHOT):
  - 21.761 seconds on high-end PC
  - 27.654 seconds on Jetson TK1
- Critical phases:
  - Matching
  - Consistency Grouping
  - Hypotheses Verification
  - Normal estimation
  - Description

Optimizations

Optimizations
CPU optimizations

- Organized normal estimation
  - Integral Images
  - Avoids expensive neighbor search operations
- High-dimensionality optimized k-d tree
  - kdTreeIndex from FLANN
  - Points have 3 dimensions
  - Descriptors have a large number (SHOT-352)
- Multi-threaded matching
  - Custom OpenMP implementation
- Multi-threaded descriptor extraction
  - PCL implementation with OpenMP
- Bounding box based on HCI information
  - 40x40x40 cm size
  - The user points to the object's region
  - The bounding box is placed in that region to extract a ROI

- CUDA parallel implementations for GPU:
  - Parallel cloud projection (depth to color)
  - Parallel bilateral filter
  - Parallel normal estimation
  - Parallel cloud resolution computation

Optimizations performance study

Table: Runtimes and speedups for the CPU optimized phases.

Optimization	Runtime (s) Original	Runtime (s) Optimized	Speedup
Efficient k-d tree	14.462	3.624	$\sim$**4x**
Multi-threaded matching	3.624	1.399	$\sim$**2.5x**
Multi-threaded descriptor	1.979	0.722	$\sim$**2.8x**
Normal estimation	2.180	0.173	$\sim$**12.3x**
Bounding box + All	27.654	7.738	$\sim$**3.6x**

- **Full pipeline runtime reduced from** ~ 27.6 **to** ~ 7.7 **seconds (**$\sim$**3.6x speedup)**

Optimizations performance study
GPU optimizations results

Table: Runtimes and speedups for the GPU optimized phases.

Phase	Runtime (s) CPU	Runtime (s) GPU	Speedup
Cloud projection	0.0345	0.0009	**∼38x**
Normal estimation	0.3496	0.0066	**∼53x**
Bilateral filter	0.1357	0.0176	**∼8x**
Cloud resolution	0.5830	0.0102	**∼57x**
All four phases	1.1027	0.0353	**∼31x**

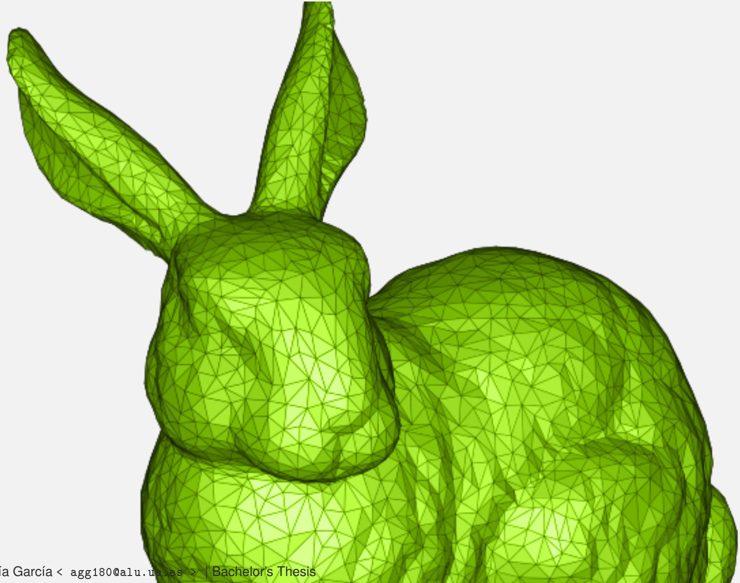
- **Runtime of the optimized phases reduced from** ~ 1.1 **to** ~ 0.035 **seconds (∼31x speedup)**
- **Full pipeline runtime reduced to** ~ 6.7 **seconds**

Conclusions

Conclusions
Highlights

- **A 3D object recognition pipeline with PCL**
  - Multi-model
  - Hypothesis verification
  - Full pose estimation and alignment
- **A 3D object reconstruction tool and a 3D dataset**
- **A extensive descriptor and pipeline experimentation**
- **Optimized CPU implementation**
  - High-dimensionality optimized k-d tree
  - Normal estimation for organized clouds
  - Multi-threaded descriptor extraction and matching
  - Bounding-box clipping based on HCI information
- **GPU implementation deployed on Jetson TK1**
  - Parallel cloud creation
  - Parallel normal estimation
  - Parallel bilateral filter
  - Parallel resolution computation

- **Implementation of keypoint detectors**
  - Further detector experimentation
- **Implementation of other local surface descriptors**
  - Mian tensor
  - TriSI
- **GPU parallelization of complex phases**
  - Descriptor extraction
  - Matching
  - Correspondence grouping
  - Hypotheses verification
- **Optimize GPU implementation to our target**
  - Use specific features: zero-copy
  - Use generic features: shared-memory, streams...



Bachelor’s Thesis

Towards a real-time 3D object recognition pipeline on GPGPU computing platforms using low-cost RGB-D sensors

Alberto García García

< agg180@alu.ua.es >

Supervisors:

José García Rodríguez
Sergio Orts Escolano

Department of Computer Technology (DTIC)
University of Alicante

Thanks for your time

Dataset