3D Camera for Mobile Device

Deborah Cohen & Dani Voitsechov
Supervisor : Raja Giryes

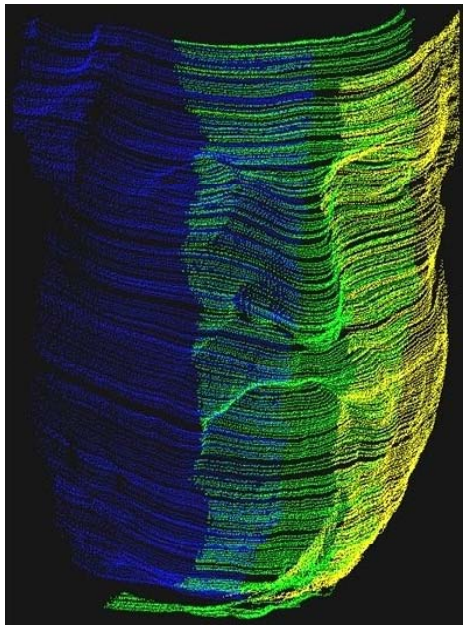
Israeli Embedded Conference
12/7/11

Outline

- Introduction
- 3D Reconstruction Algorithm
- Mobile System Structure
- Results

What is 3D imaging?

- Collect data on a real-world object
- Analyze the collected data
- Construct 3D image



Why 3D imaging?

Medical Imaging

Entertainment

Autonomous navigation system

Security applications

Active stereometric scanner

[*Bronstein, Bronstein, Gordon & Kimmel* '03]

Passive stereo

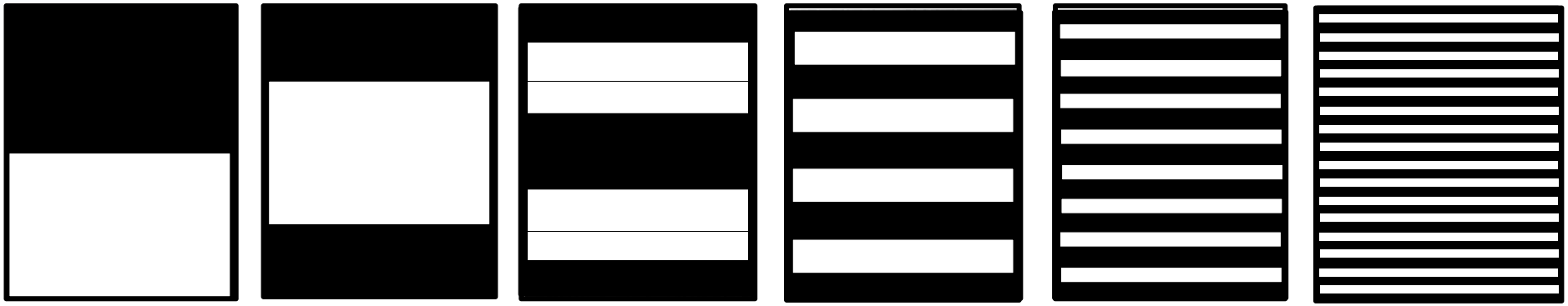
Active stereo

Active stereometric scanners

Structured light projections

Time-multiplexed patterns

Active stereo hierarchy



Project definition and goals

Creating a mobile 3D camera system:

- System requirements:
  - ❑ Mobile
  - ❑ Stand-alone
  - ❑ Automatic
- Objectives:
  - ❑ Real-time
  - ❑ Accuracy
  - ❑ Low cost

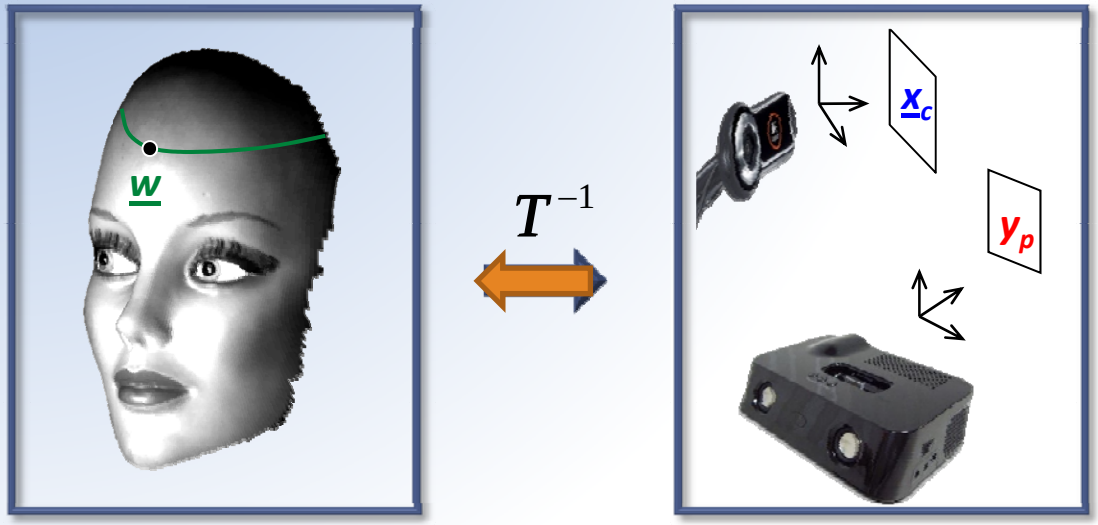
Existing system:
standard camera, projector, PC

Suitable prototype for mobile devices

Outline

- Introduction
- 3D Reconstruction Algorithm
- Mobile System Structure
- Results

Projective model



Object plane

Camera/Projector planes

$\underbrace{x_c, y_c}_{\underline{x}_c}$ – camera coordinates

y_p – projector coordinate

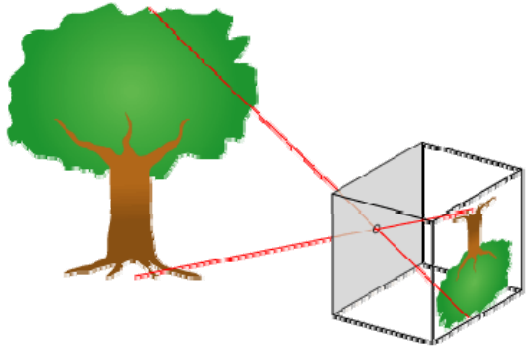
$\underbrace{x_w, y_w, z_w}_{\underline{w}}$ – world coordinates

- Forward projection (linear) $T:(x_w, y_w, z_w) \rightarrow (x_c, y_c, y_p)$
- Backward projection $(x_w, y_w, z_w) = T^{-1}(x_c, y_c, y_p)$

Projection matrices

$$\left.\begin{aligned} \underline{x}_c &= \underline{\underline{C}}_c \underline{w} \\ \underline{y}_p &= \underline{\underline{C}}_p \underline{w} \end{aligned}\right\} T$$

- Based on pin-hole model:



- Composed of:
  - o Camera/projector parameters (focal distance…)
  - o Rotation matrix
  - o Translation vector

Algorithm phases

Phase 1 – Calibration (performed once)

- Determining the projection matrices $\underline{\underline{C}}_c$ and $\underline{\underline{C}}_p$

Phase 2 – Decoding

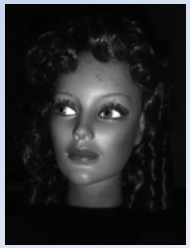
- Computing the projection coordinate y_p

Phase 3 – Reconstruction

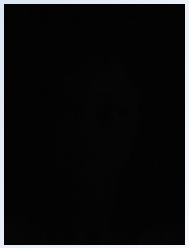
- Finding the world coordinates $\underline{w}$

Decoding

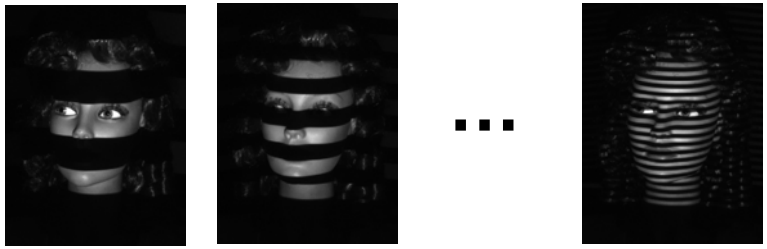
- Input:



illuminated image



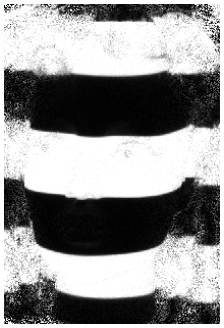
not illuminated image

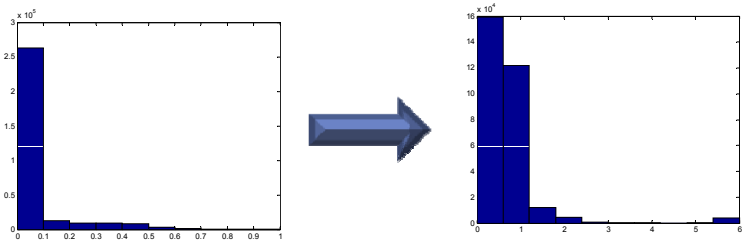


N coded light patterns

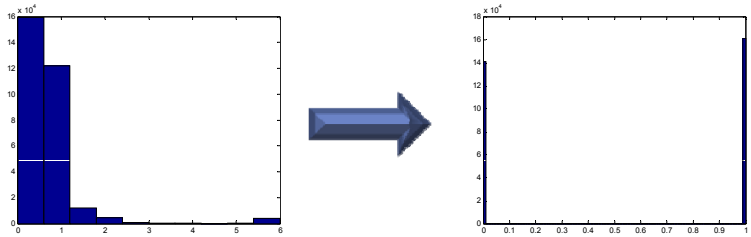
- Preprocessing:

1. Normalization:





2. Thresholding:



Decoding (cont.)

$$y_p = \sum_{k=1}^{N} 2^{N-k} B_k(x, y)$$

B_k – k^{th} binary pattern, $1 \le k \le N$

Projected coded light patterns

- Output :

  y_p – projector coordinate for each pixel

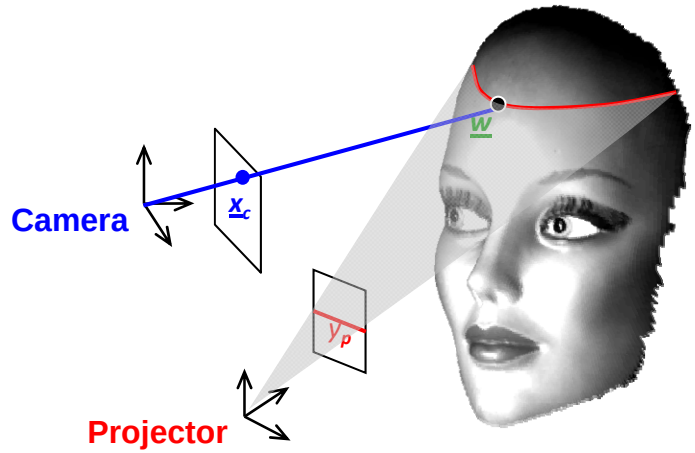
Reconstruction

Input : $\underline{\underline{C}}_c, \underline{\underline{C}}_p$ – projection matrices

x_c, y_c, y_p – projector and camera coordinates

- Using the back projection:

$$\left(x_w, y_w, z_w\right) = T^{-1}\left(x_c, y_c, y_p\right)$$



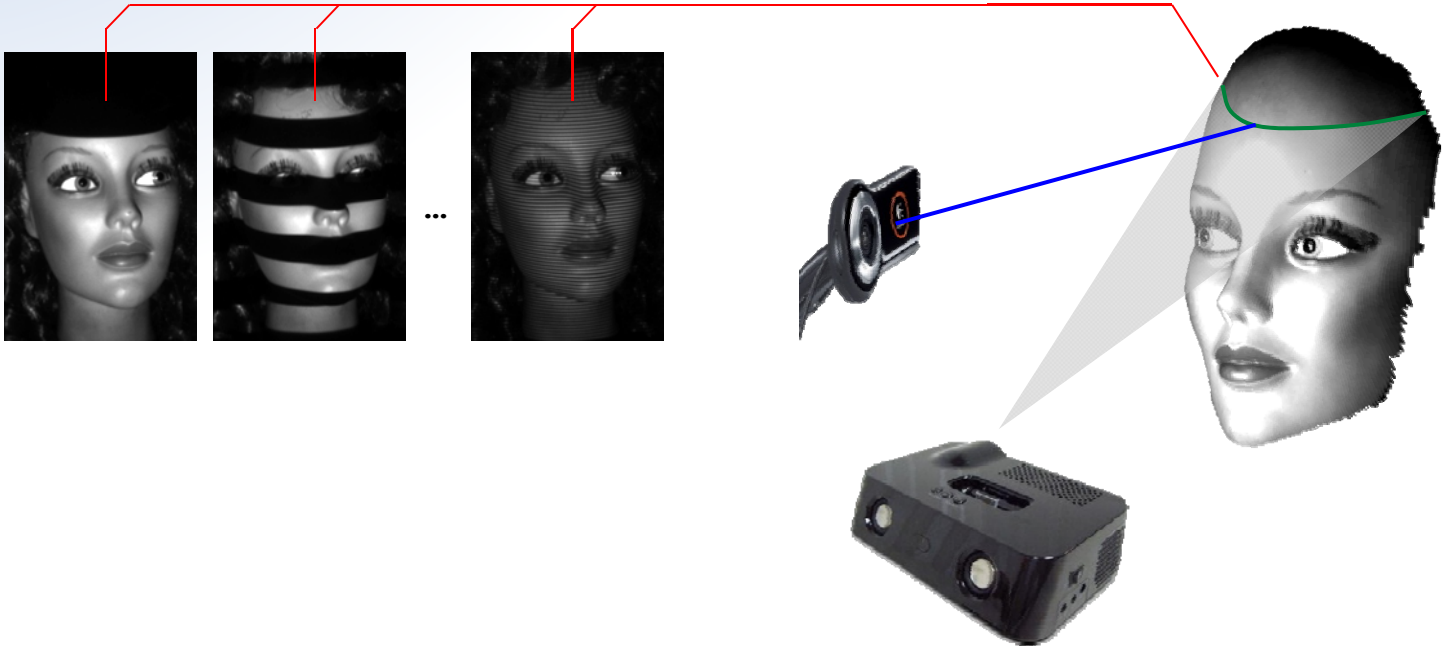
- Output : x_w, y_w, z_w – world coordinates

Outline

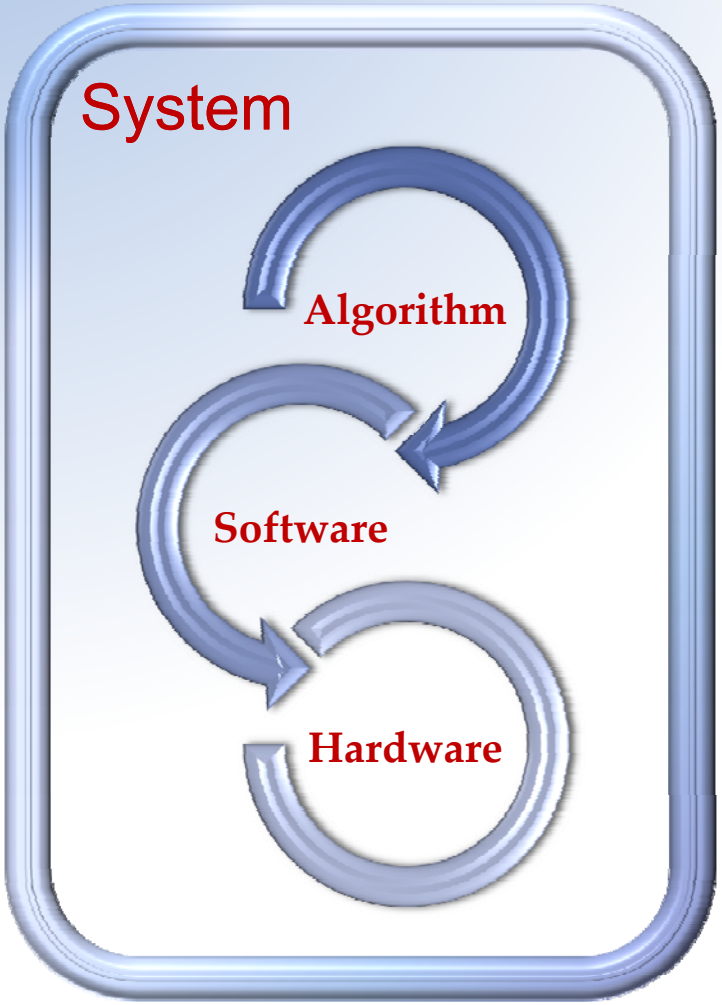
- Introduction
- 3D Reconstruction Algorithm
- **Mobile System Structure**
- Results

Structured light scanner

- Projecting time multiplexed patterns
- 2D image acquisitions
- 3D reconstruction



System implementation

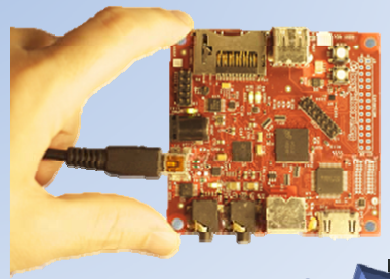


- Adaptations due to limited hardware

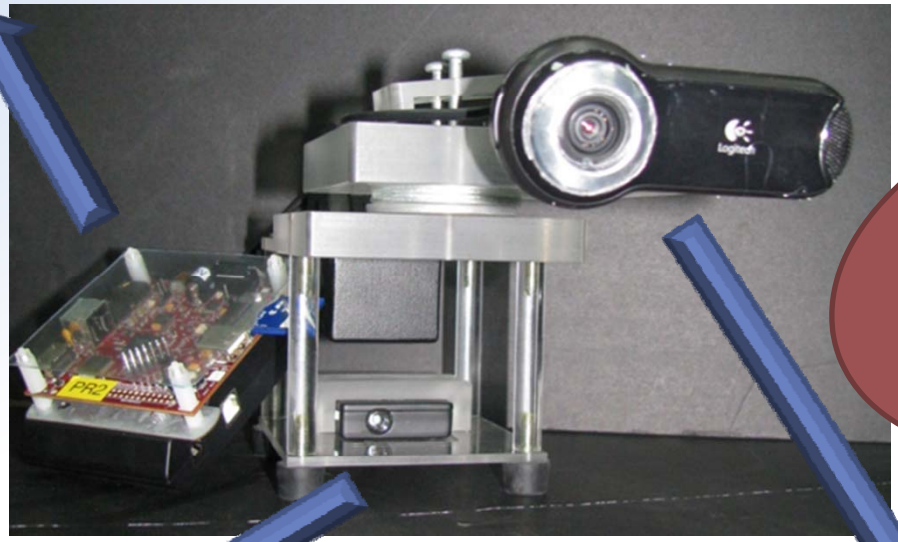
- Fixed point
- Task division between three processors
- Driver modifications

- BeagleBoard© beagleboard.org
- OMAP 3530 TI
- I/O devices: Camera and Projector

System



Beagleboard©

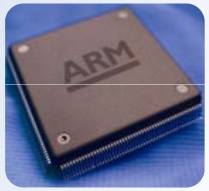


Device built in the EE workshop

TI DLP Pico Projector

Logitech QuickCam Pro 9000

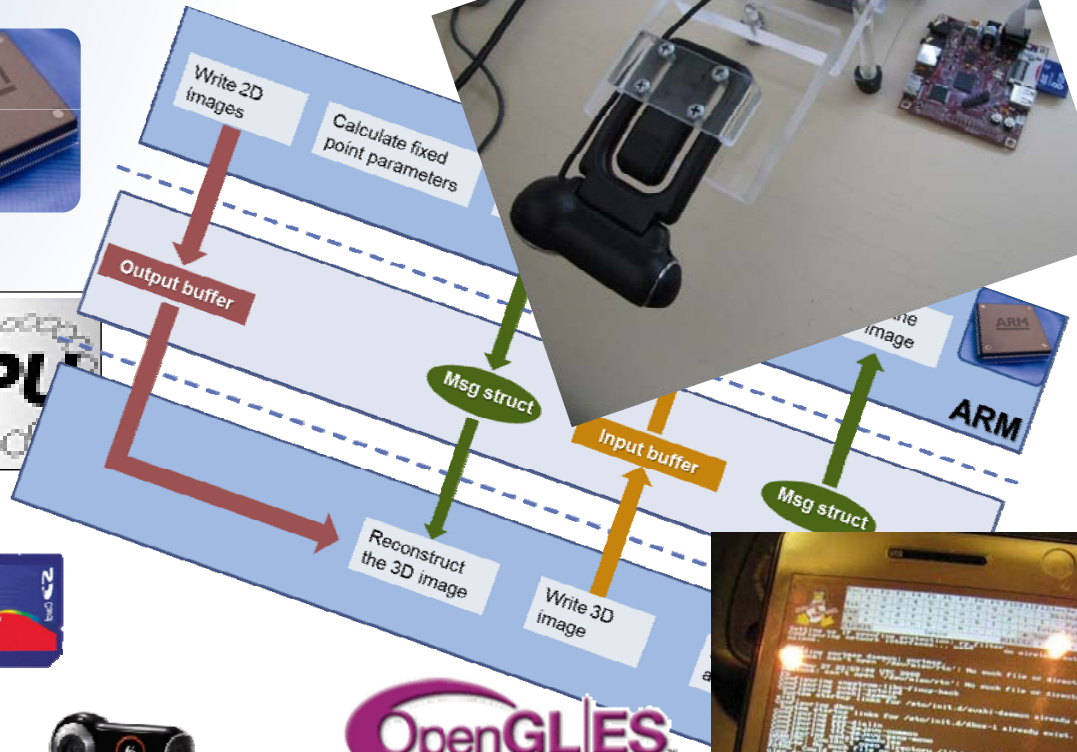
Challenges

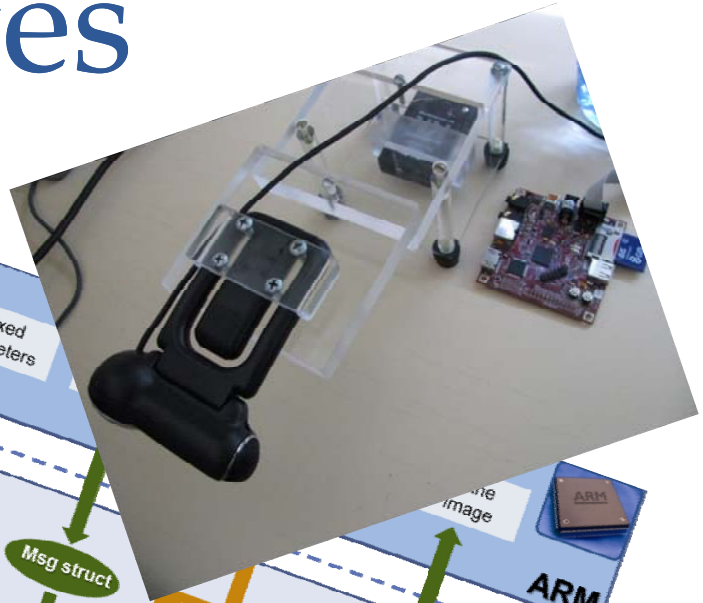

ARM

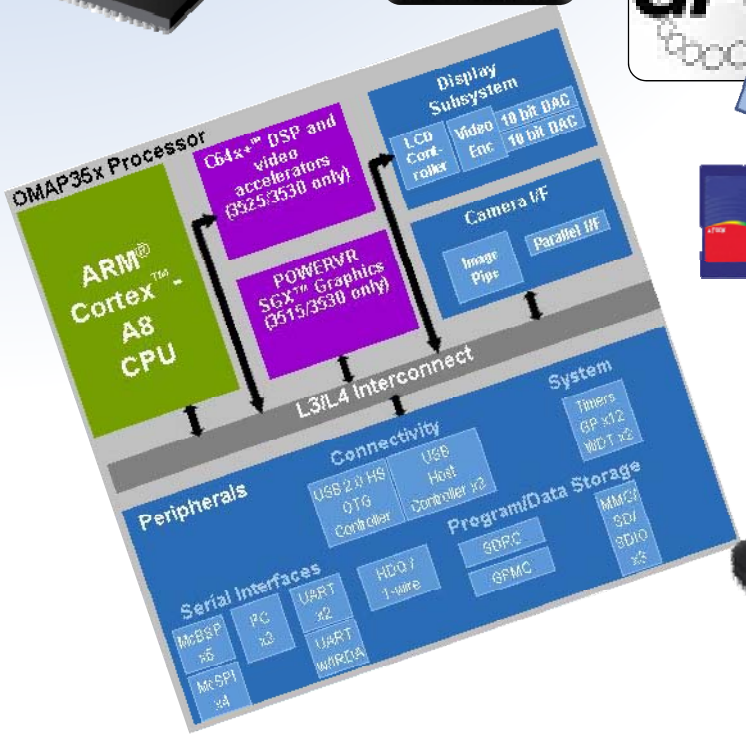

OMAP3530
Texas Instruments


TI Third Party Network
Member
DSP
TEXAS INSTRUMENTS

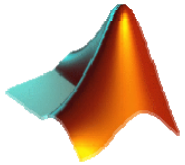

GPU


Write 2D images
Calculate fixed point parameters
Output buffer
Msg struct
Input buffer
Msg struct
Reconstruct the 3D image
Write 3D image
ARM

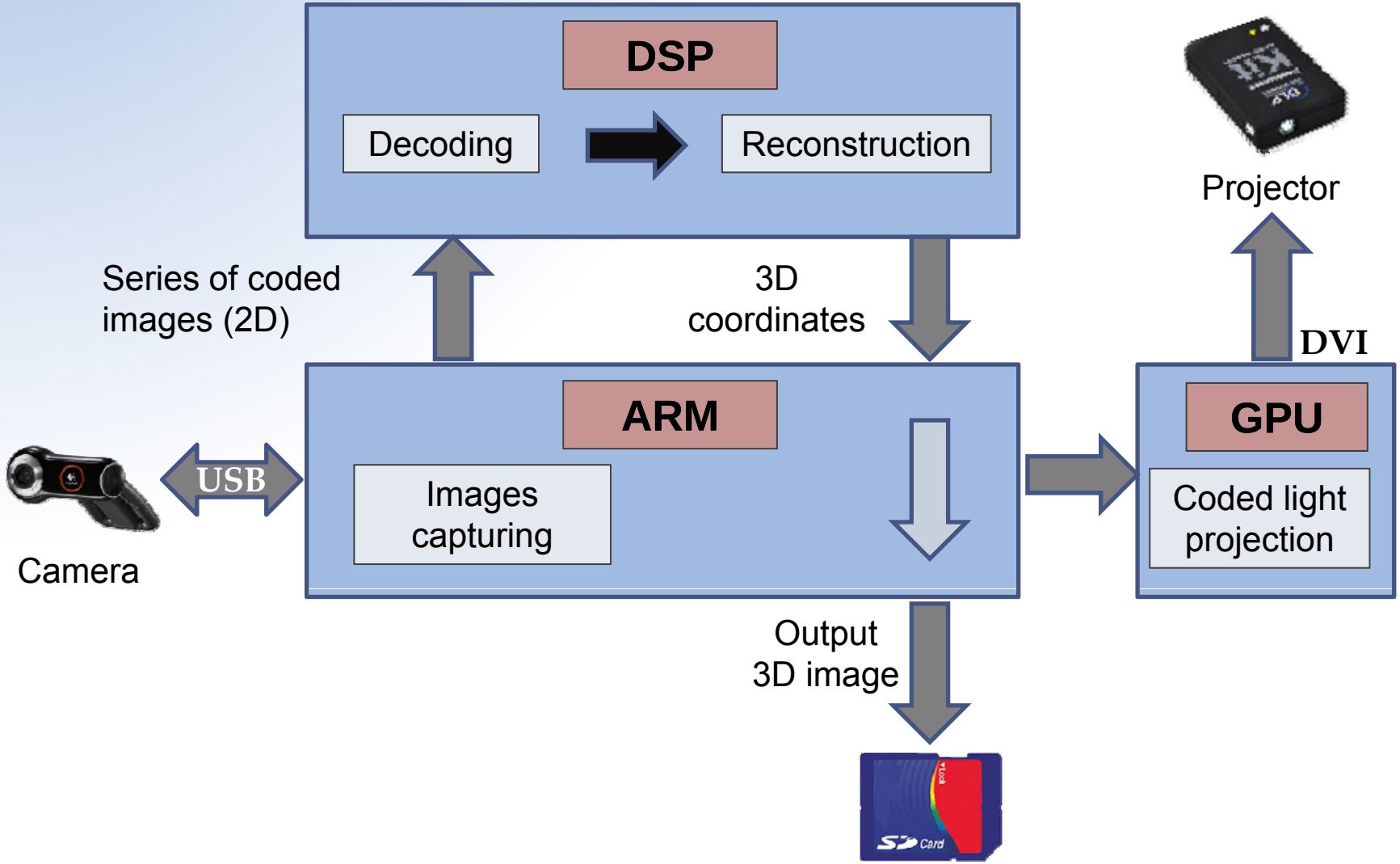



OMAP35x Processor
ARM® Cortex™ - A8 CPU
C64x+™ DSP and video accelerators (3525/3530 only)
POWERVR SGX™ Graphics (3515/3530 only)
Display Subsystem
LCD Cont-roller
Video Enc
10 bit DAC
10 bit DAC
Camera I/F
Image Pipe
Parallel I/F
L3/L4 Interconnect
System
Timers GP x12 WDT x2
Connectivity
USB 2.0 HS OTG Controller
USB Host Controller x2
Peripherals
Program/Data Storage
MMC/SD/SDIO x3
SDRC
GPMC
HDQ / 1-wire
Serial Interfaces
McBSP x5
I²C x3
UART x2
UART w/IrDA
McSPI x4


OpenGL|ES

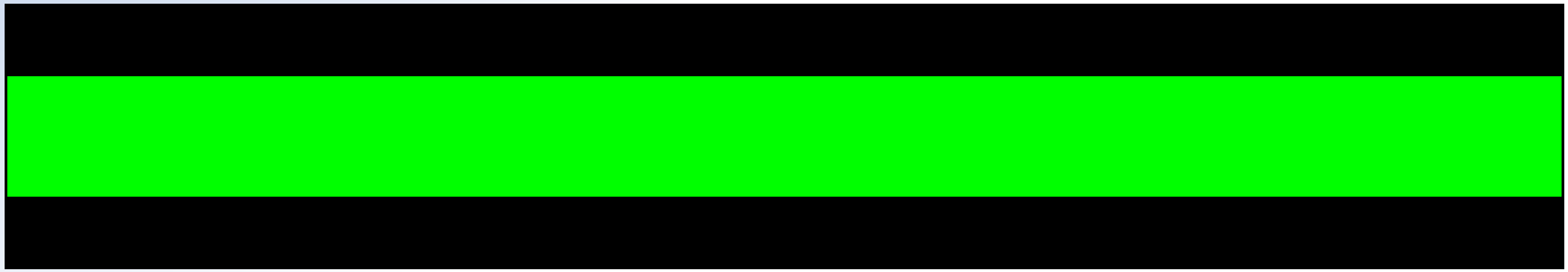


Task division



Implementation steps

I. Patterns projection: 7 alternating green and black stripes



II. Image capture: storing intensity image

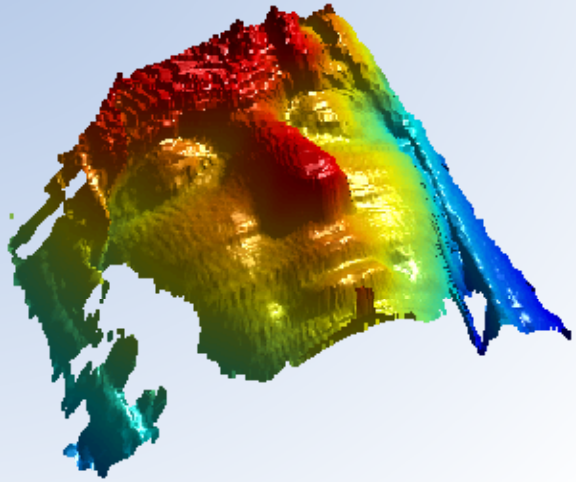
III. 3D reconstruction: performed in the DSP

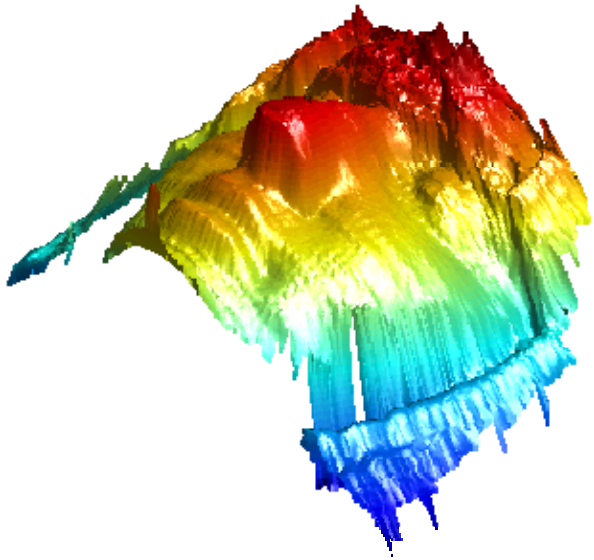
Outline

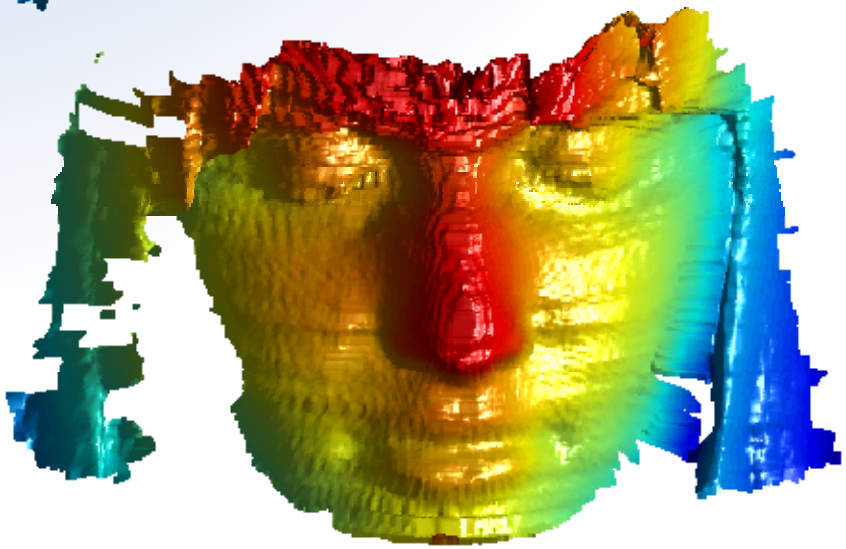
- Introduction
- 3D Reconstruction Algorithm
- Mobile System Structure
- Results

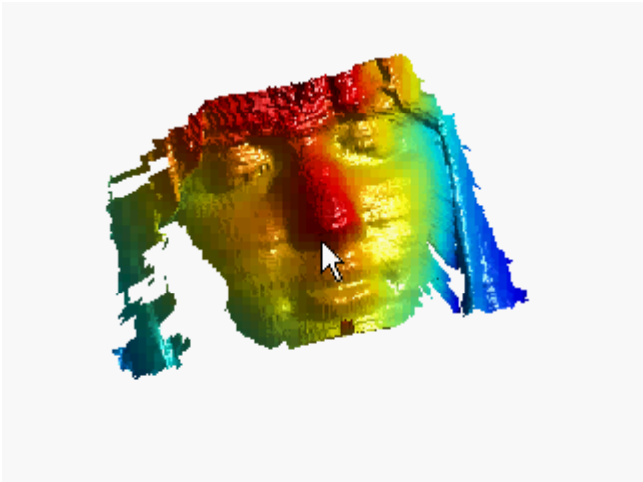
Demo

Results



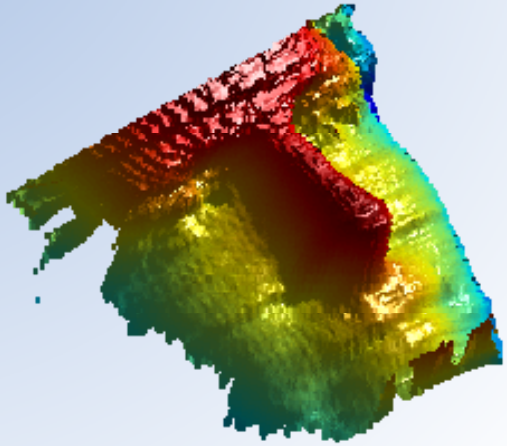


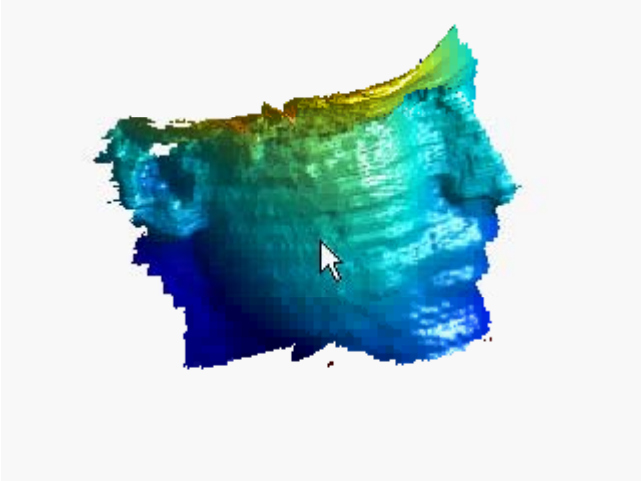


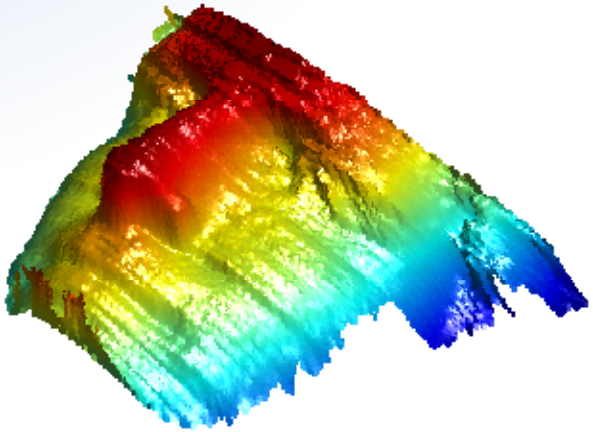


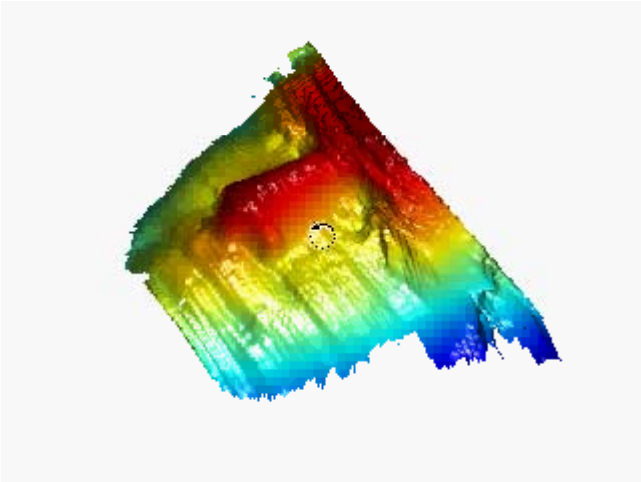
Deborah

Results



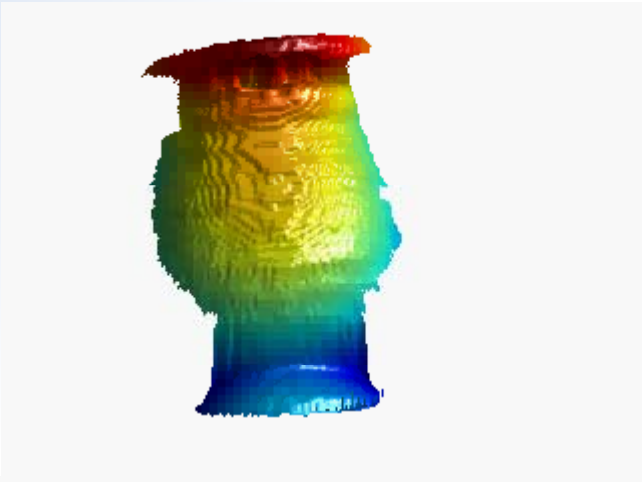


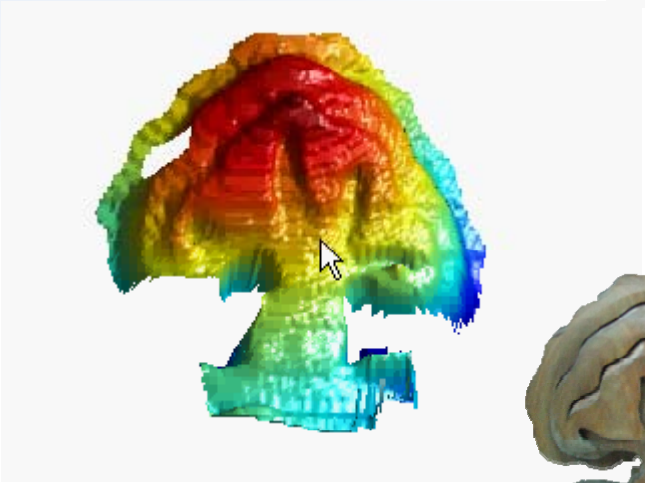


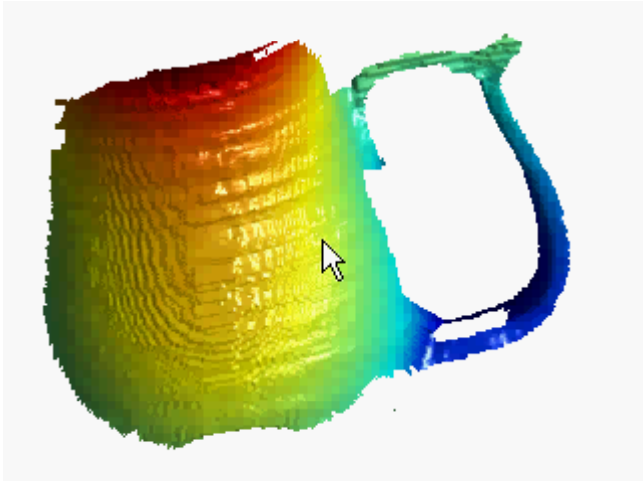


Dani

Results







Performance analysis

- Accuracy: **RMS = 0.50 - 1 [mm per pixel]**
- Complexity: **O(*#pixels*· *#patterns*)**

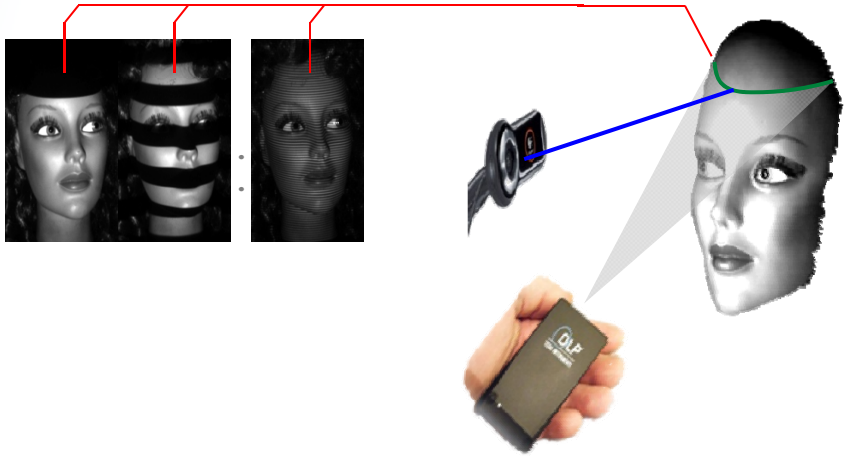
TASK	EFFECTIVE TIME (sec)
DSP reconstruction	**0.24**
ARM reconstruction (for comparison)	1.19

TASK	EFFECTIVE TIME (sec)
Acquisition	2.81 (*)

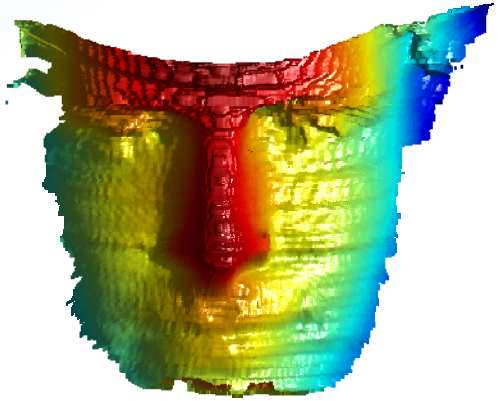
(*) Multiple projections (software sync)

Summary

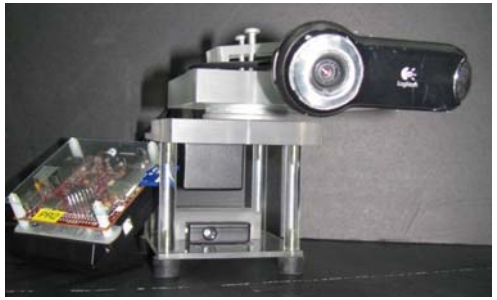
Suitable solution for mobile devices



Active stereometric scanner



3D reconstructed face



System prototype

Achievements

Miniaturization	• Mobile system
Accuracy	• Accurate 3D reconstruction
Real time reconstruction	• 4 reconstructions per second
Low cost	• Common devices
Low power	• Energy efficient devices

Thanks

Pavel

Ori

Avi

Raja