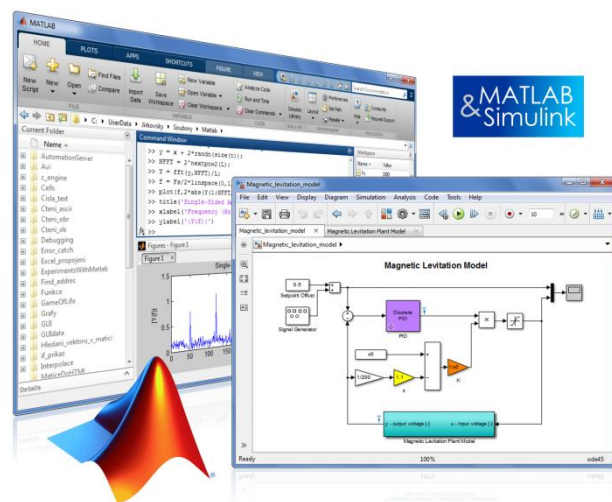


MATLAB & Simulink

3-D počítačové vidění a rekonstrukce prostorové scény



Jaroslav Jirkovský
jirkovsky@humusoft.cz

Počítačové vidění (Computer vision)

Využívá obrázu a videa k detekci, klasifikaci a sledování objektů nebo událostí za účelem pochopení reálné situace



Zpracování obrazu

Odstranění šumu
Nastavení kontrastu
Měření
...

Počítačové vidění

Detekce
Identifikace
Klasifikace
Rozpoznávání
Sledování
...

Interpretace

Chodec
Cyklista
Nákladní vůz
Osobní vůz
Dopravní omezení
Nehoda
...



Praktické aplikace počítačového vidění

- s využitím MATLABu



Zpracování obrazu x Počítačové vidění

- **Dvě kategorie zpracování obrazu v MATLABu**
 - klasické zpracování obrazu
 - počítačové vidění
- **Klasické zpracování obrazu**
 - úprava obrazu a odstranění šumu
 - práce s barevnými prostory
 - transformace obrazu
 - morfologické operace, prahování, tvorba masky
 - **podrobné informace: www.seminar.cz Zpracování obrazu a videa v MATLABu**
 - viz.: <http://www.humusoft.cz/wwwseminare/>
- **Počítačové vidění**
 - algoritmy zaměřené na detekci a sledování objektů, OCR
 - 3-D vidění a rekonstrukce prostorové scény

Propojení MATLABu s kamerami

- **Základní MATLAB**

- webové kamery (USB)
- IP kamery

- **Image Acquisition Toolbox**

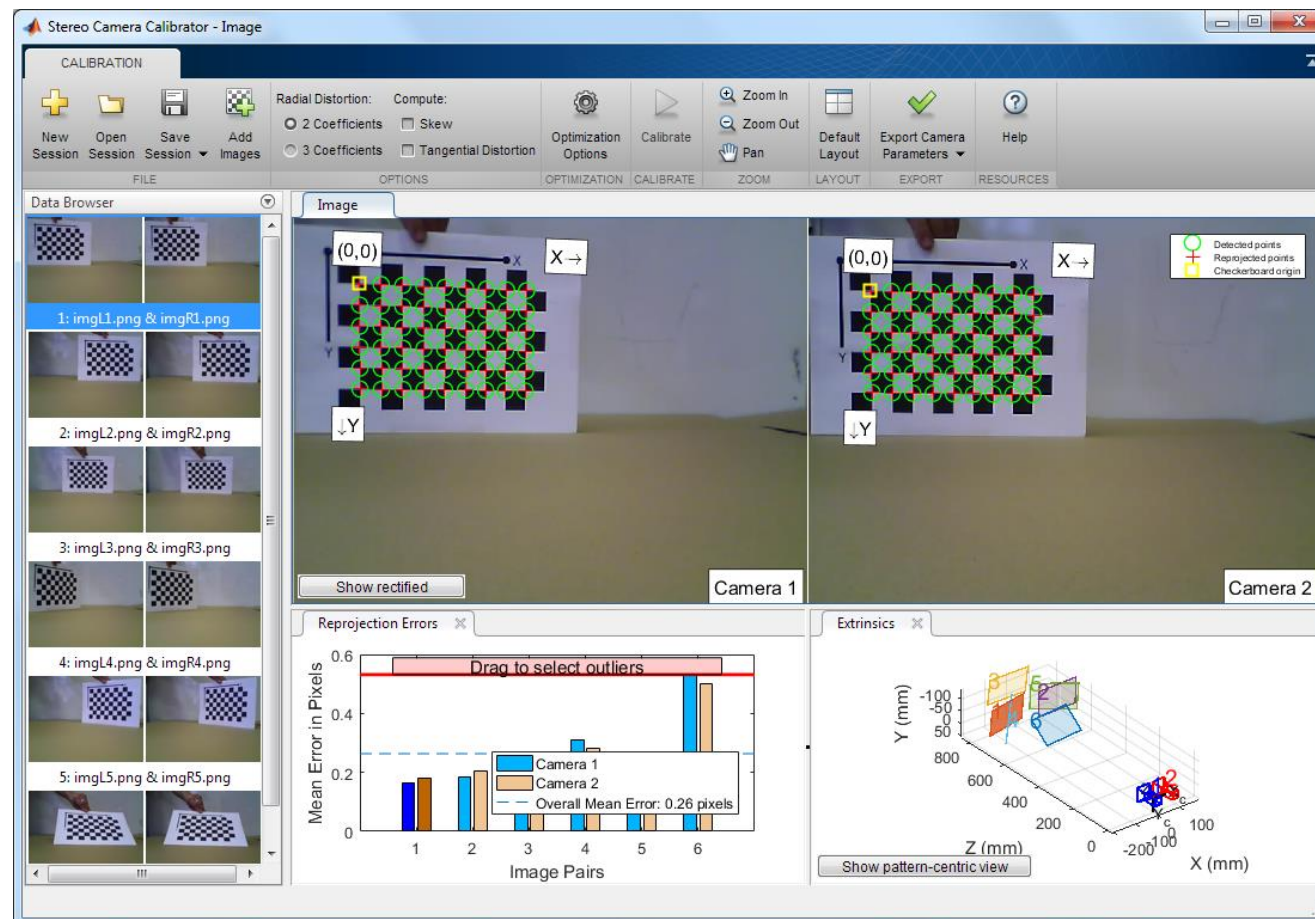
- pokročilá manipulace s kamerami

- USB3 Vision , GigE Vision, GenICam GenTL, DCAM Compatible FireWire (IIDC 1394)
- OS Generic Video Interface
- Kinect



Kalibrace dvojice kamer

- Úkol:
 - provést kalibraci kamer
- Řešení:
 - využijeme grafické rozhraní Stereo Camera Calibrator



Parametry

- Model kamery:

$$- w[x \ y \ 1] = [X \ Y \ Z \ 1] \begin{bmatrix} R \\ t \end{bmatrix} K$$

vnější parametry

vnitřní parametry

$$- K = \begin{bmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{bmatrix}$$

- Radiální zkreslení

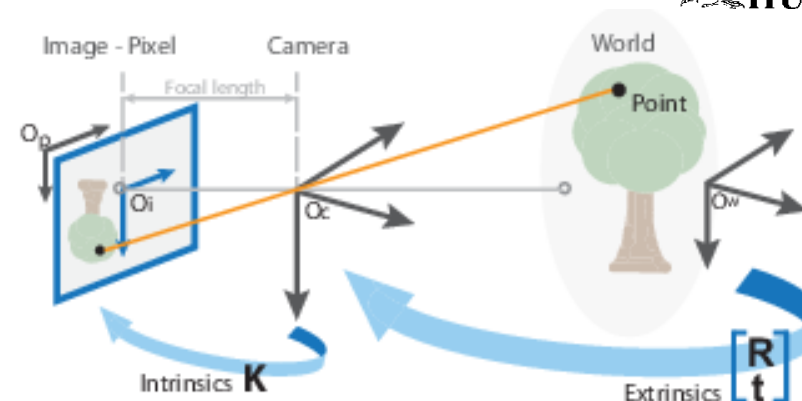
- 2 nebo 3 koeficienty

$$- x_D = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$- y_D = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

- Ize též odhadnout

- zešikmení, tangenciální zkreslení



- X, Y, Z ... reálné souřadnice
- x, y ... obrazové souřadnice
- w ... měřítko
- K ... matice vnitřních parametrů
- R ... prostorové natočení kamery
- t ... posunutí kamery

- $f_x = \frac{F}{p_x}, f_y = \frac{F}{p_y}$... [pix]
- F ... ohniskové vzdálenost [mm]
- p_x, p_y ... rozměr pixelu [mm]
- s ... zešikmení (skew) $s = f_y \tan \alpha$
- c_x, c_y ... souřadnice optického středu [pix]

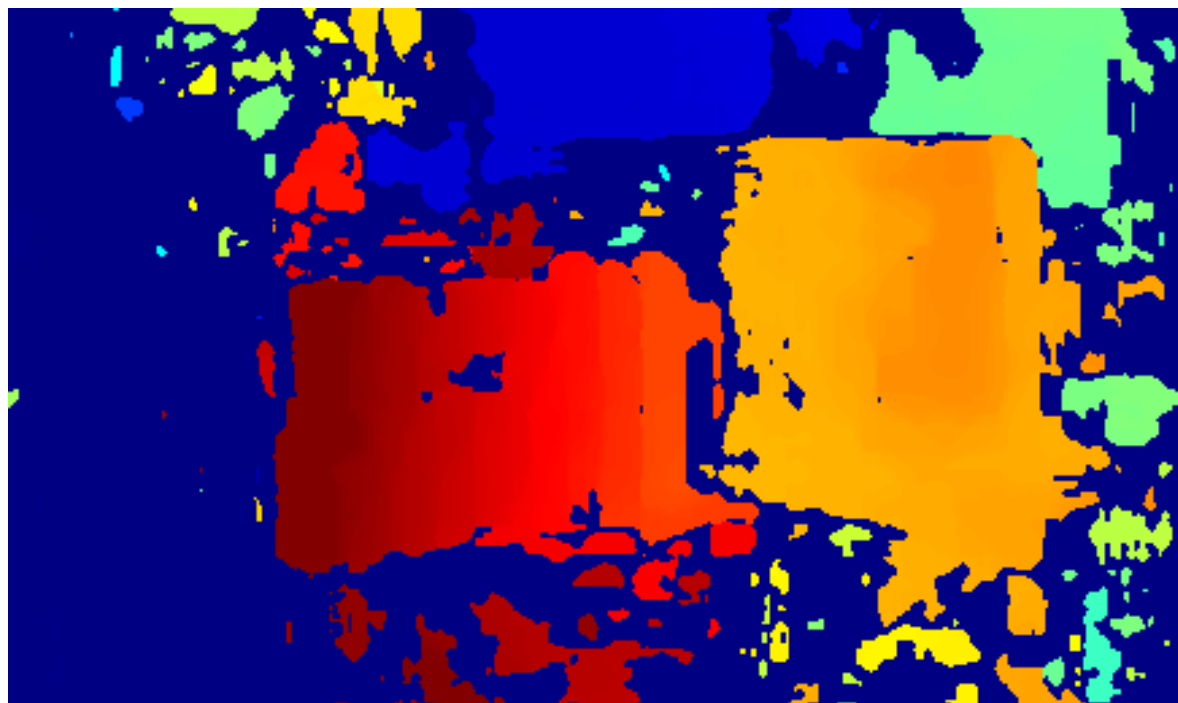
Rekonstrukce 3-D scény

- **Úkol:**
 - rekonstruovat 3-D scénu z dvojice snímků
- **Řešení:**
 - využijeme funkce Computer Vision System Toolboxu a získáme prostorové rozložení bodů ve scéně



Rekonstrukce 3-D scény - hloubková mapa

- `rectifyStereoImages()`
- `disparity()`

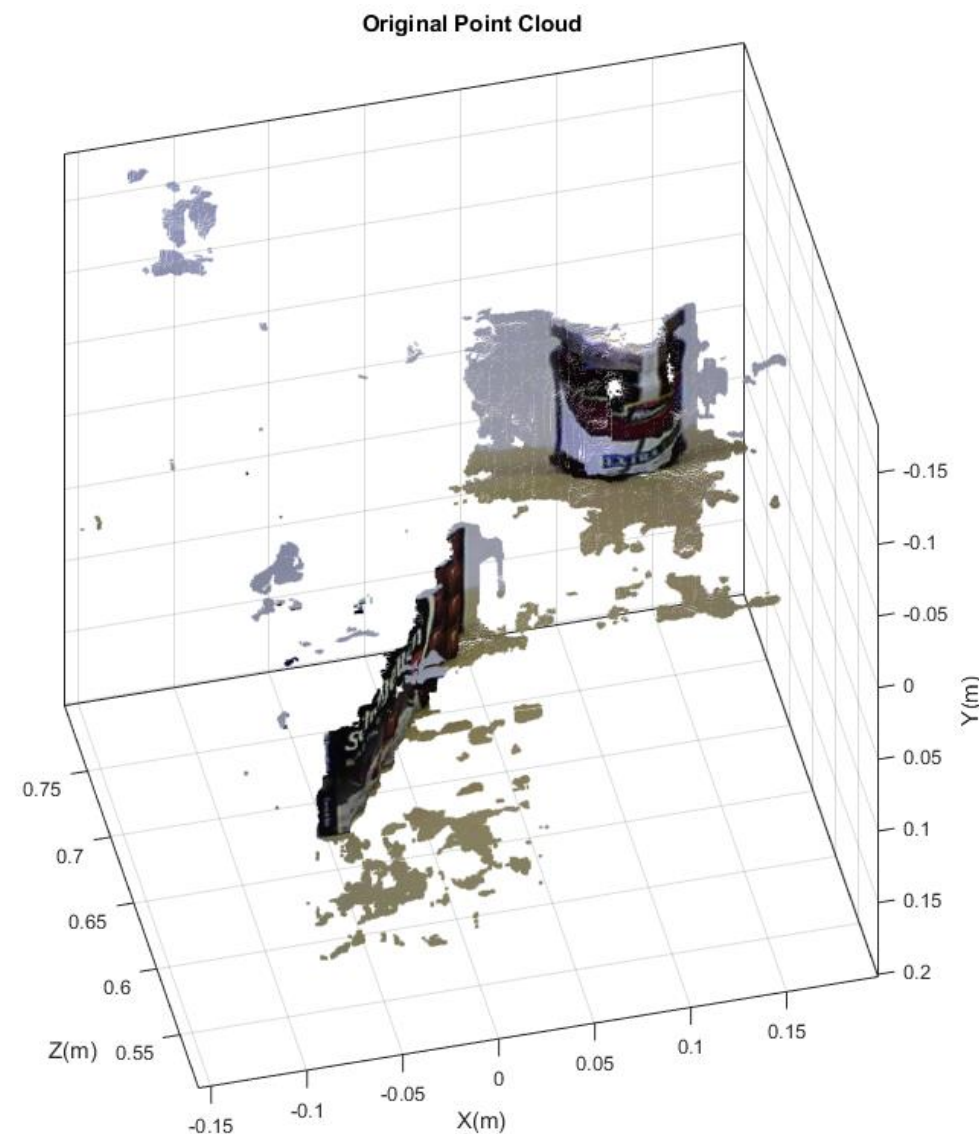


Rekonstrukce 3-D scény – point cloud

- `reconstructScene()`
- `pcshow()`

- **Další možností objektu PointCloud**
 - uložení, načtení, získání ze zařízení Kinect
 - hledání bodů v dané oblasti nebo okolí

 - odstranění šumu
 - podvzornování
 - transformace
 - registrace nebo merge dvou point cloud objektů



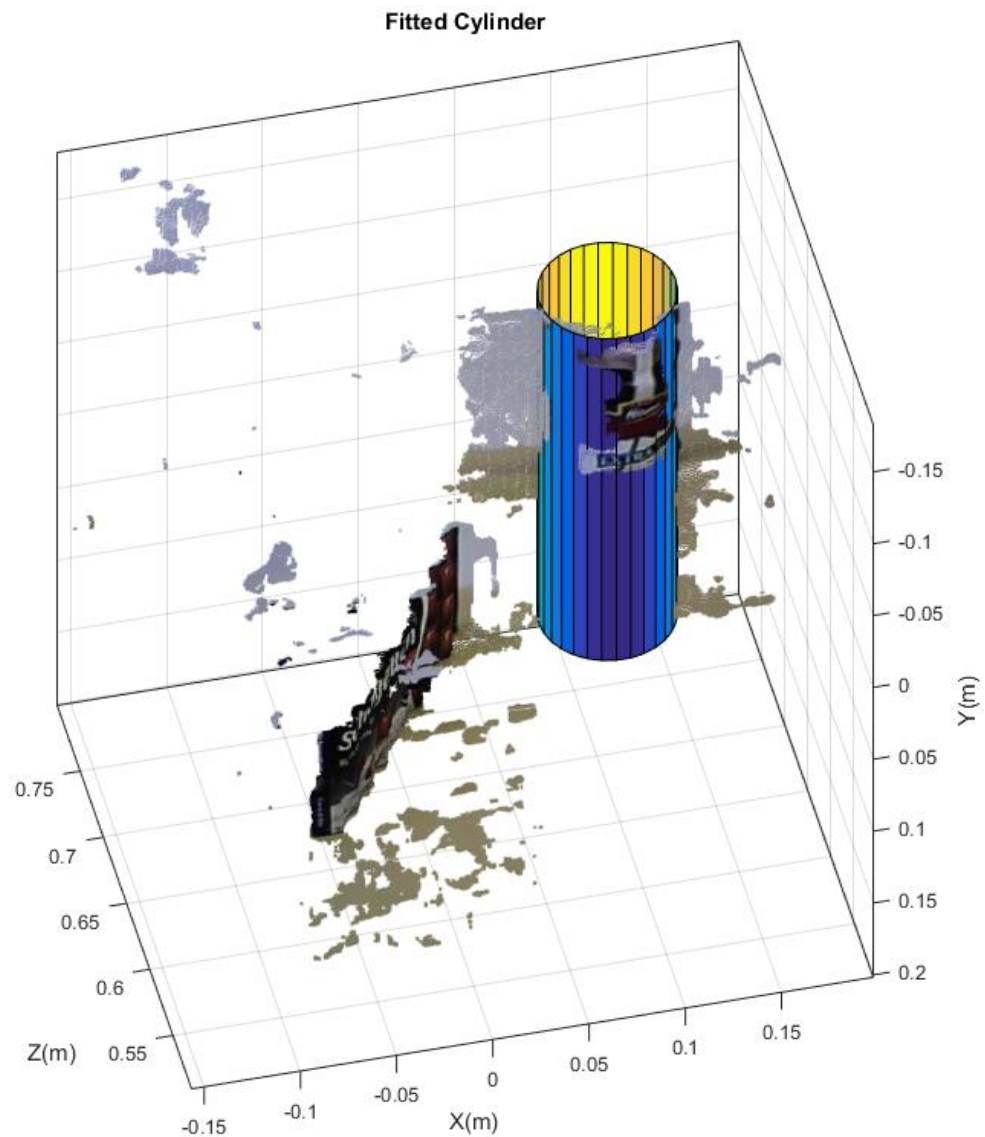
Extrakce těles z prostorové scény

- **Úkol:**
 - najít polohu skleničky v prostorové scéně
- **Řešení:**
 - využijeme funkce pro prokládání 3-D bodů geometrickými tělesy a extrahujeme body v okolí nalezeného tělesa



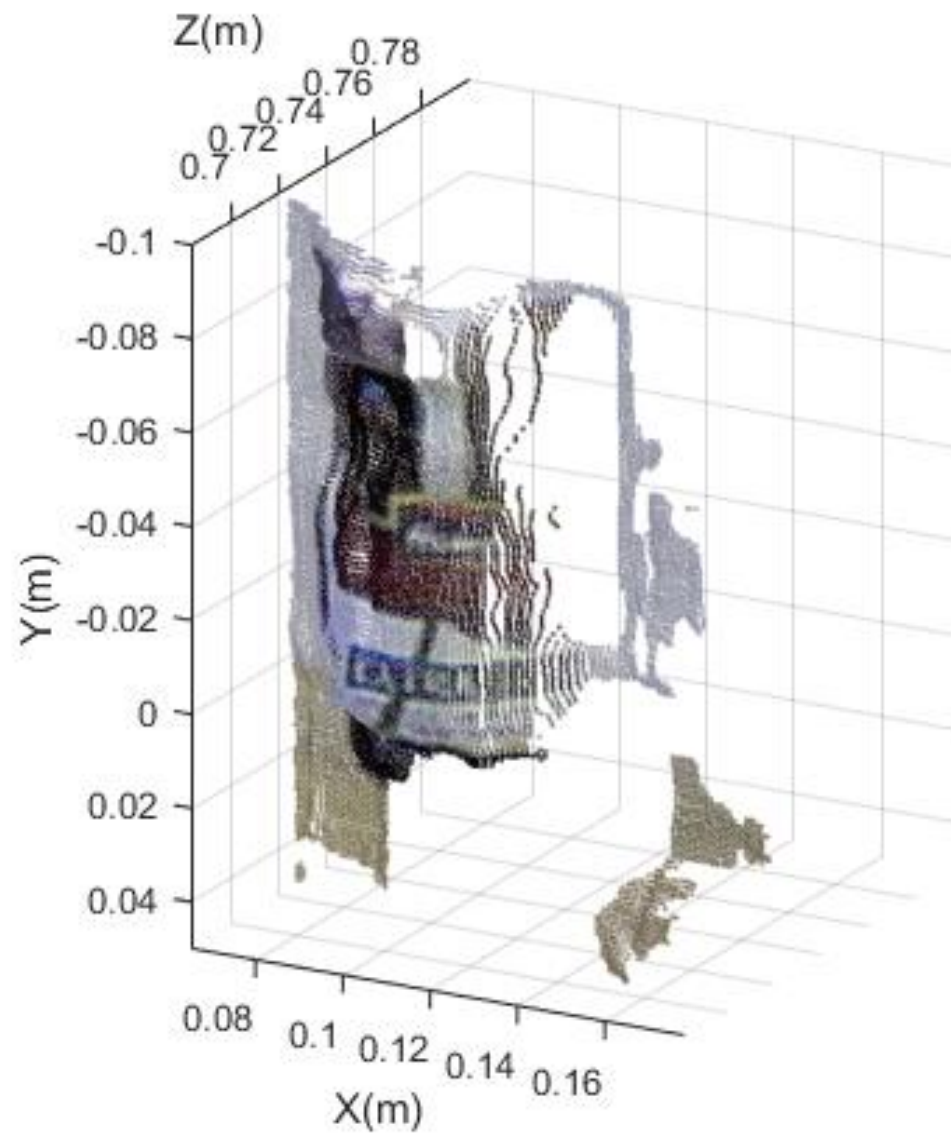
Prokládání 3-D bodů geometrickými tělesy

- `pcfitcylinder()`
- `pcfitsphere()`
- `pcfitplane()`



Extrakce tělesa

- `select()`



Poloha objektu

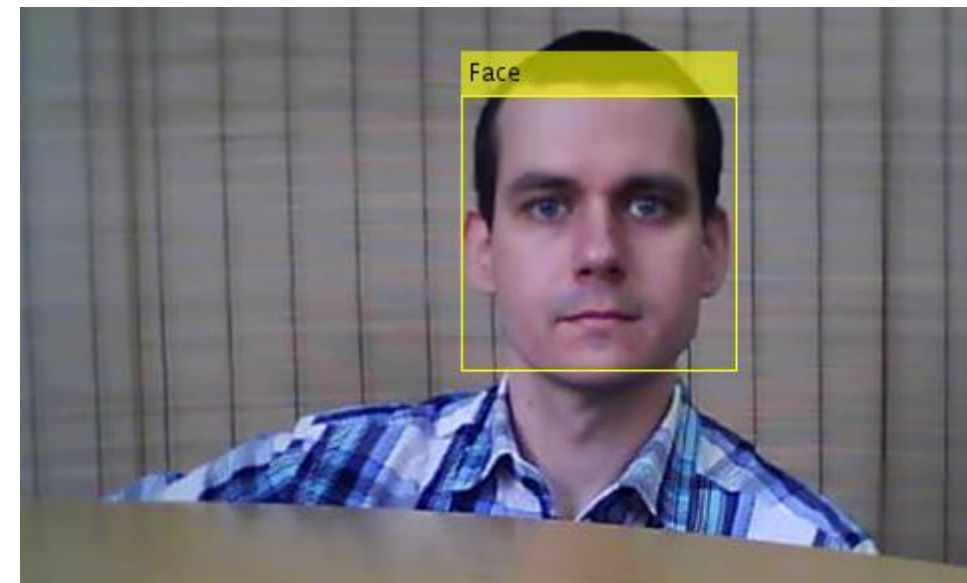
- **Úkol:**
 - najít polohu obličeje v prostorové scéně
- **Řešení:**
 - využijeme funkce pro detekci obličeje a triangulaci



Detekce obličeje

- **Detekce částí lidského těla**

- `vision.CascadeObjectDetector()`
- před-trénovaný klasifikátor
- obličej, nos, oči, ústa, horní polovina těla

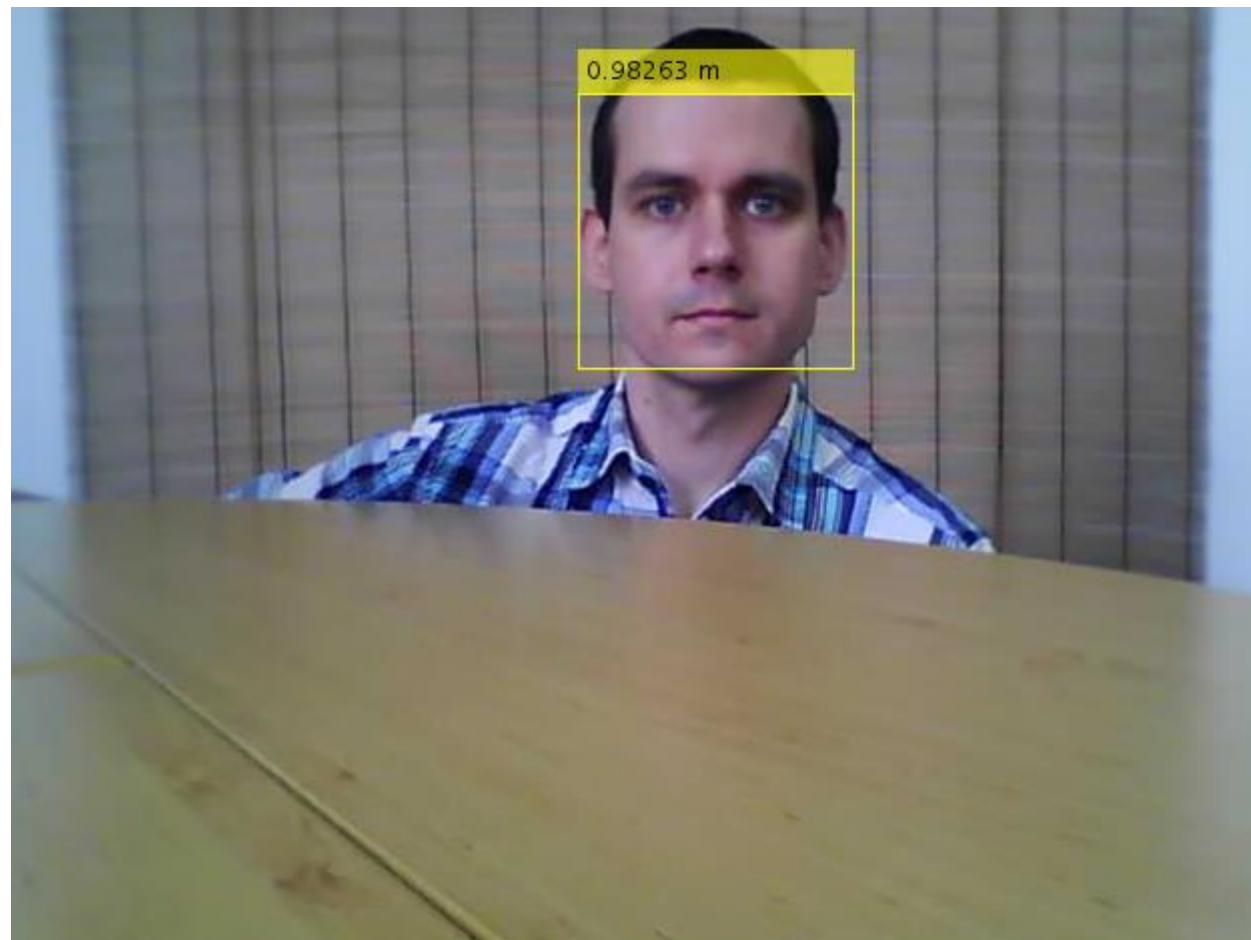


- **Možnost trénování uživatelského klasifikátoru**

- `trainCascadeObjectDetector()`

Poloha obličeje

- `triangulate()`



Detekce objektů

- Pomocí shodných příznaků
 - rohové body, SURF, MSER, BRISK



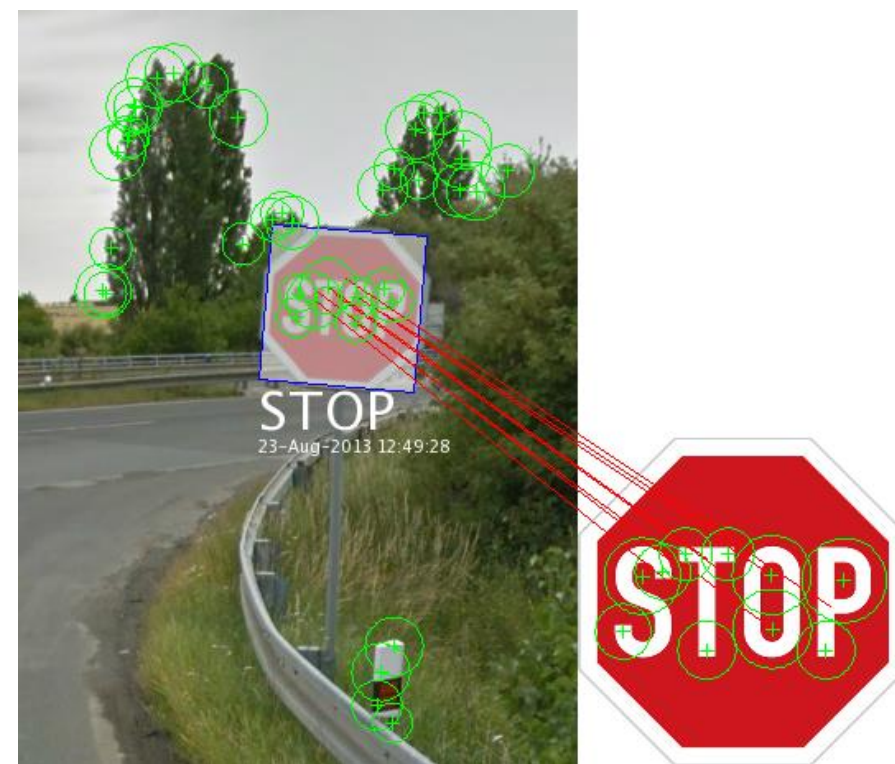
- detect Features(), extractFeatures(),
- matchFeatures(), estimateGeometricTransform()

- Bag of Visual Words

- bagOfFeatures, imageCategoryClassifier

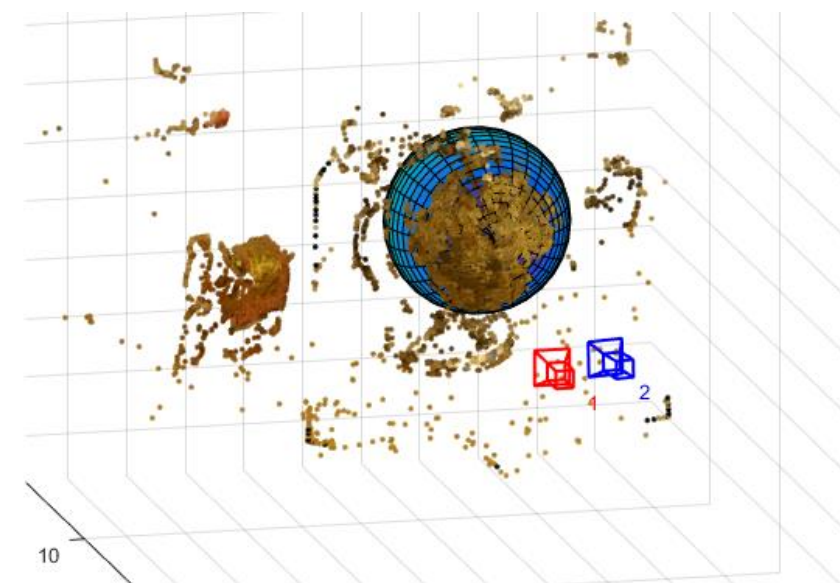
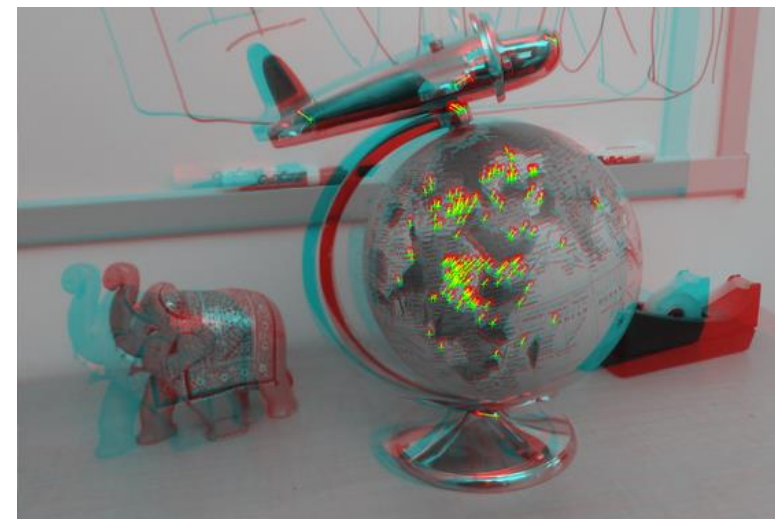
- Sledování objektů

- vision.PointTracker, vision.HistogramBasedTracker, vision.ForegroundDetector



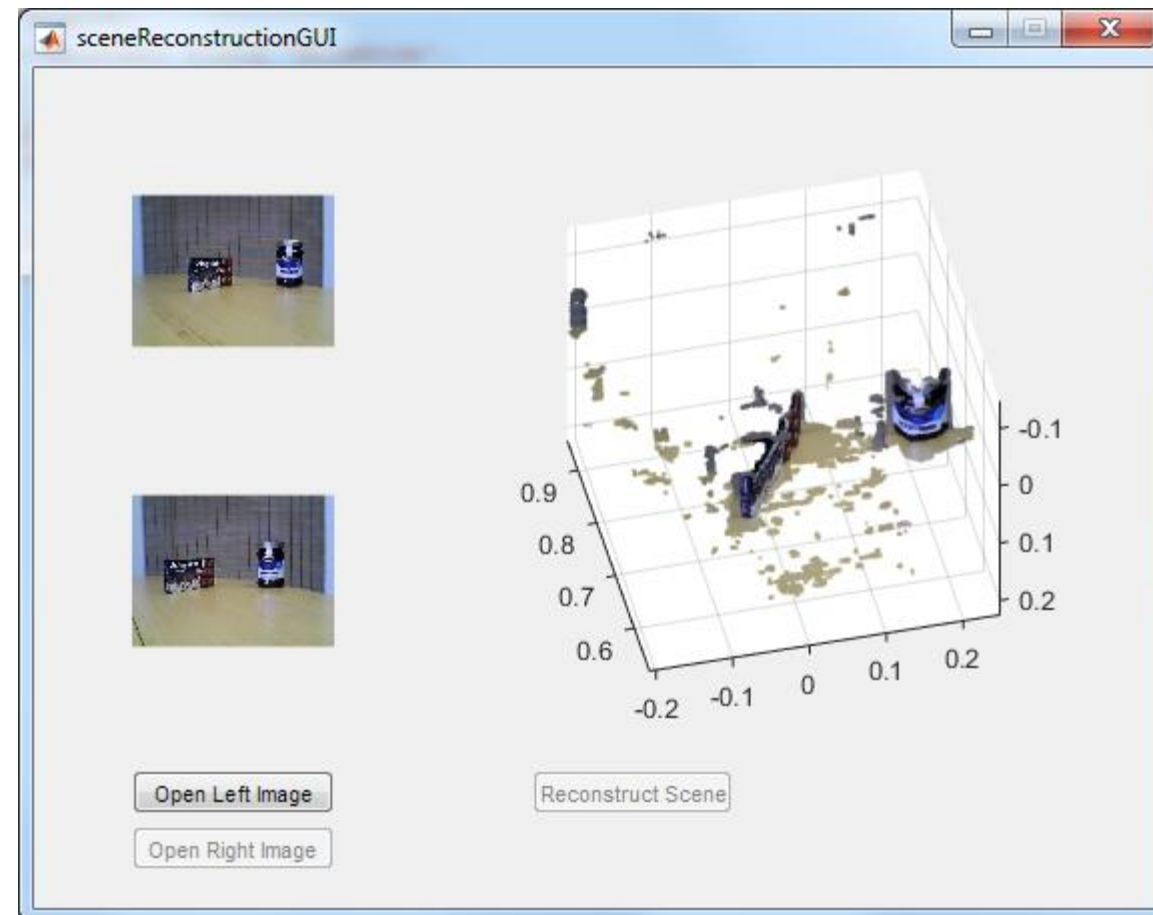
Structure From Motion

- Postup využívající předchozích metod
- Ze dvou pohledů
- Z několika pohledů
- Postupy popsány v dokumentaci
 - volba vhodných nástrojů pro různé podmínky
 - ukázkové příklady

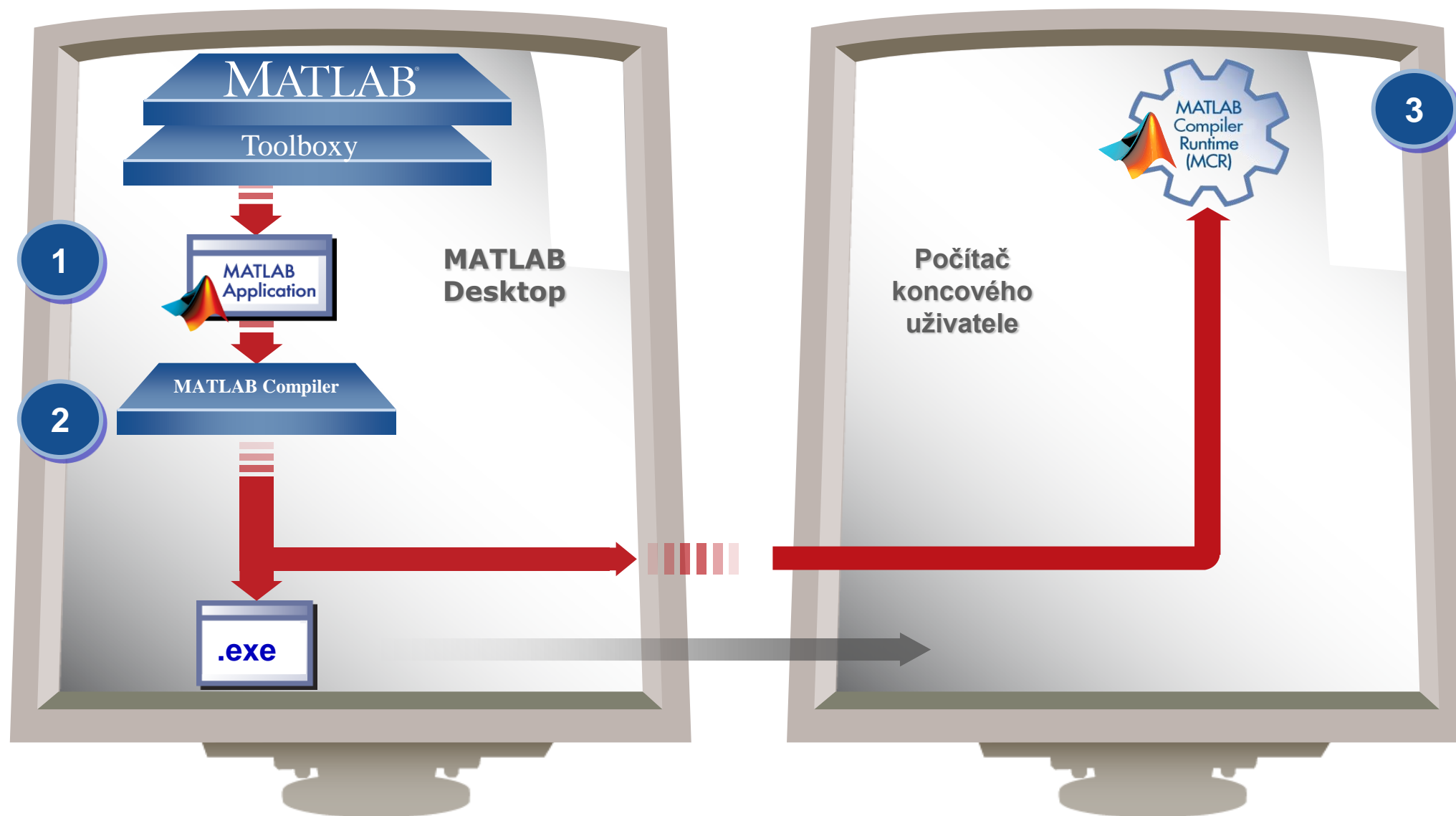


Grafické uživatelské rozhraní

- **Funkce je možné začlenit do uživatelského GUI**
 - jednodušší práce pro jiné uživatele
- **Samostatně spustitelná aplikace**
 - MATLAB Compiler



Princip fungování samostatně spustitelné aplikace



Otázky ...

... a odpovědi