Augmenting Reality, Naturally:

Scene Modelling, Recognition and Tracking with Invariant Image Features

by

Iryna Gordon

in collaboration with David G. Lowe
Laboratory for Computational Intelligence
Department of Computer Science
University of British Columbia, Canada

the highlights

computer vision

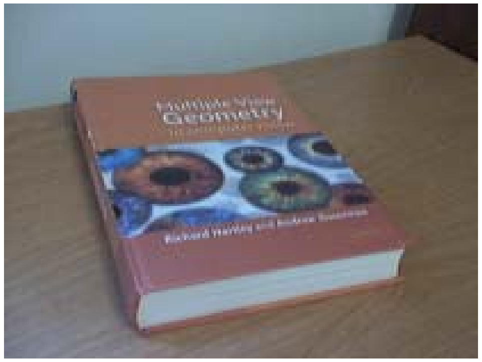
automation:

- acquisition of scene representation
- camera auto-calibration
- scene recognition from arbitrary viewpoints

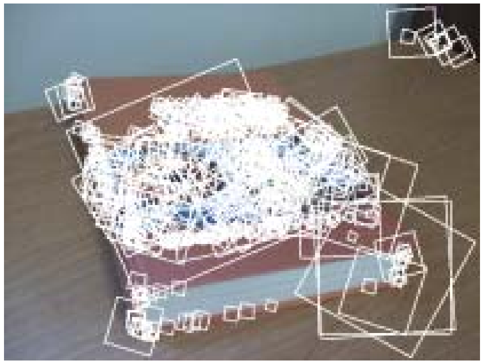
versatility:

- easy setup
- unconstrained scene geometry
- unconstrained camera motion
- distinctive natural features

natural features



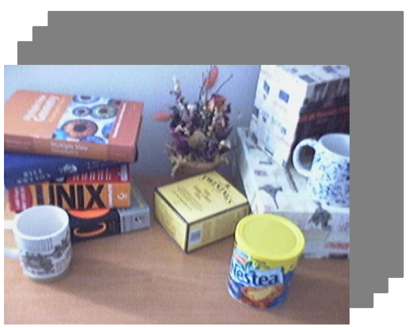
Scale Invariant
Feature Transform
(SIFT)



- characterized by image location, scale, orientation and a descriptor vector
- invariant to image scale and orientation
- partially invariant to illumination & viewpoint changes
- robust to image noise
- highly distinctive and plentiful

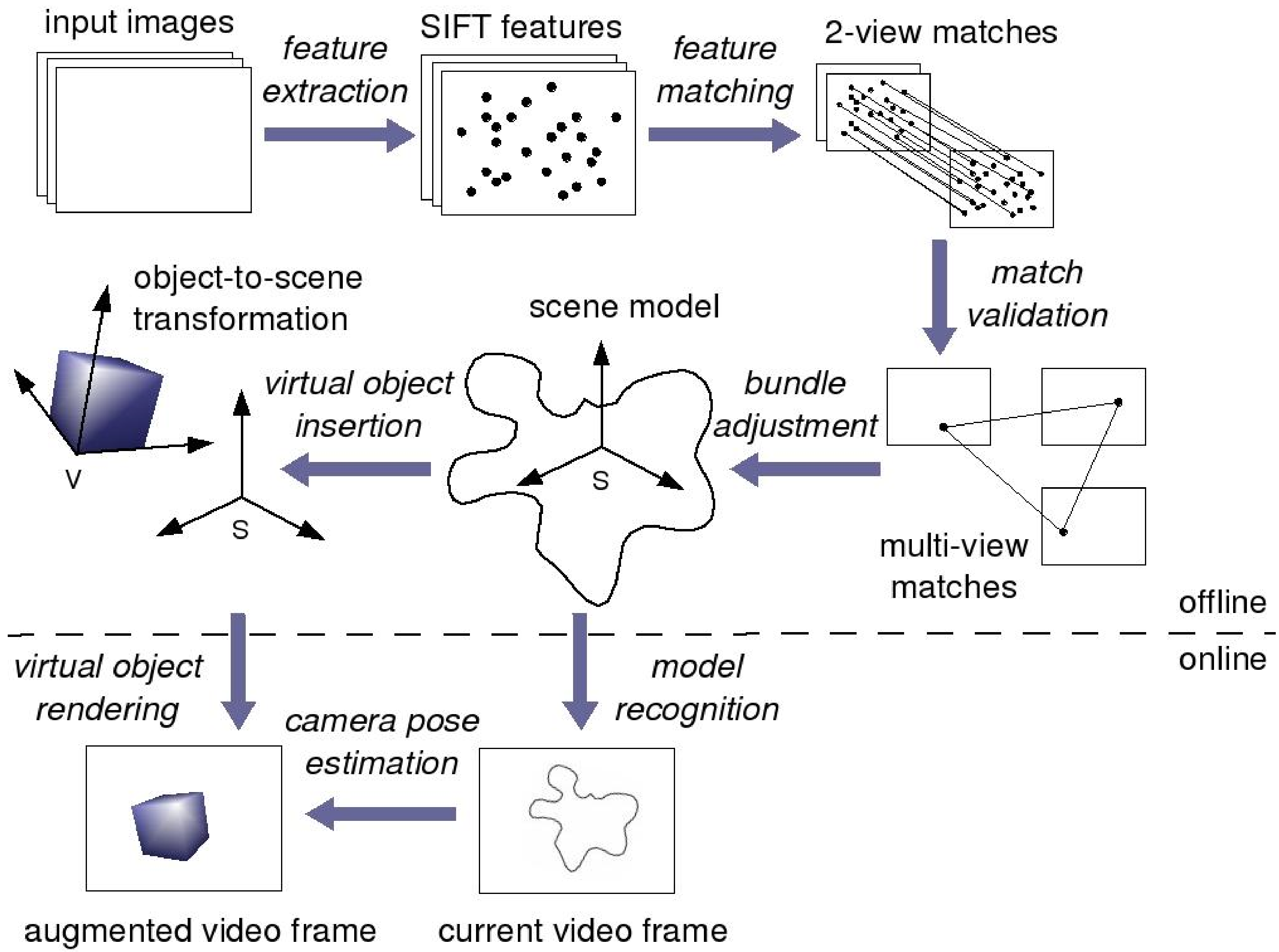
David G. Lowe. Distinctive image features from scale-invariant keypoints.
International Journal of Computer Vision, 2004.

what the system needs



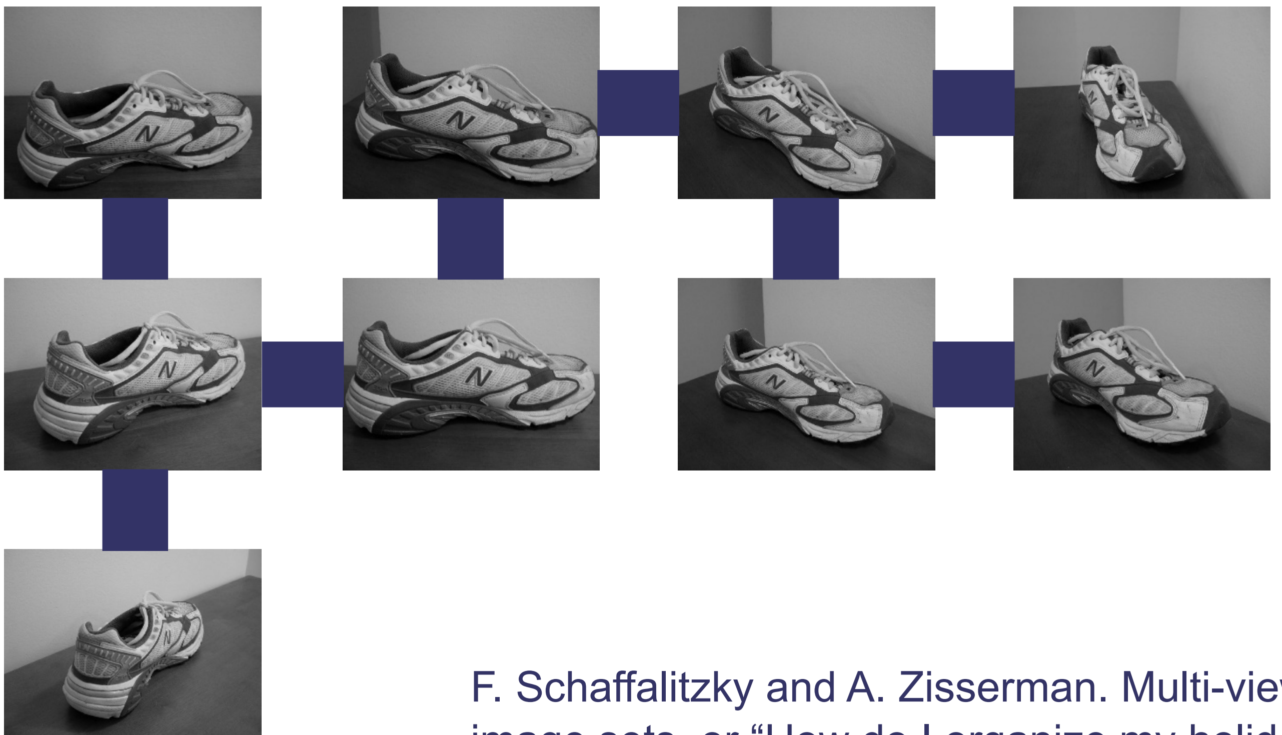
- computer
- off-the-shelf video camera
- set of reference images:
    - unordered
    - acquired with a handheld camera
    - unknown viewpoints
    - at least 2 images

what the system does



modelling reality: feature matching

- best match – smallest Euclidean distance between descriptor vectors
- 2-view matches found via Best-Bin-First (BBF) search on a k-d tree
- epipolar constraints computed for N -1 image pairs with RANSAC
- image pairs selected by constructing a spanning tree on the image set:



F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or “How do I organize my holiday snaps?”. *ECCV,* 2002.

modelling reality: scene structure

- Euclidean 3D structure & auto-calibration from multi-view matches via direct **bundle adjustment**:

$$\min_{\mathbf{a}_{ij}} \sum_i \sum_j \left\| w_j \left(\mathbf{x}_{ij} - \tilde{\mathbf{x}}_{ij} \right) \right\|^2$$

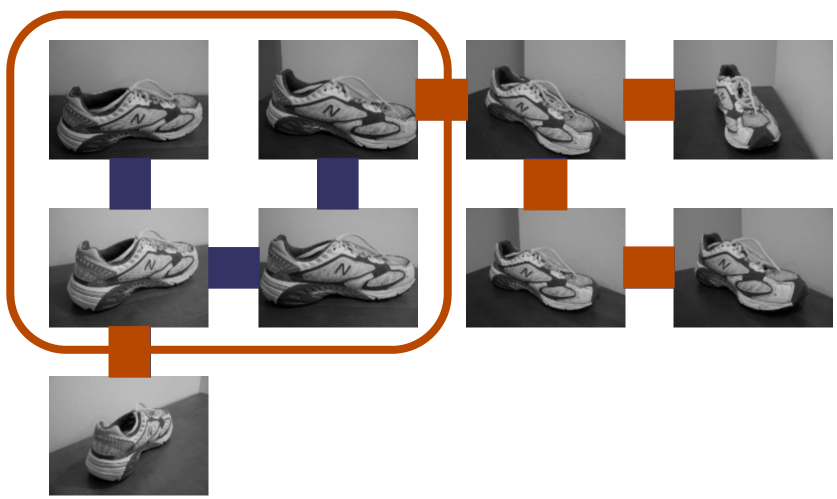
$$\mathbf{x}_{ij} = \Pi\left(R_i \mathbf{X}_j + \mathbf{t}_i \right) = \Pi\left(\left[X_j', Y_j', Z_j' \right]^T \right) = \begin{bmatrix} f \frac{X_j'}{Z_j'} + p_u \\ af \frac{Y_j'}{Z_j'} + ap_v \end{bmatrix}$$

R. Szeliski and Sing Bing Kang. Recovering 3D shape and motion from image streams using non-linear least squares. Cambridge Research, 1993.

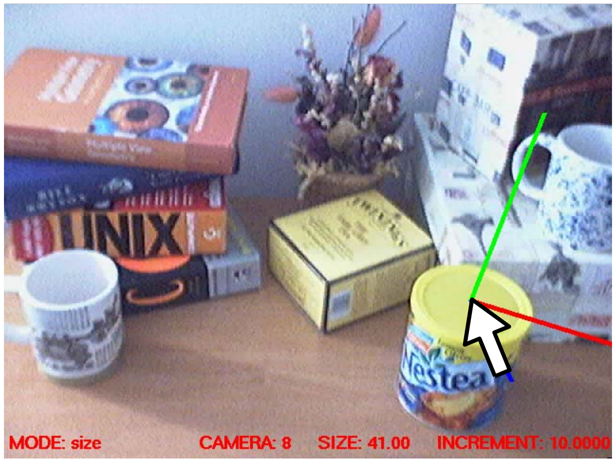
modelling reality: an improvement

- Problem:
  - computation time increases exponentially with the number of unknown parameters
  - trouble converging if the cameras are too far apart (> 90 degrees)

- Solution:
  - select a subset of images to construct a partial model
  - incrementally update the model by resectioning and triangulation
  - images processed in order automatically determined by the spanning tree



modelling reality: object placement



initial placement in 2D

determining relative depth

adjusting size and pose

rendered object in reference images

camera pose estimation

- model points' appearances in reference images are stored in a k-d tree
- 2D-to-3D matches $\left(\tilde{\mathbf{x}}_{tj}, \mathbf{X}_j\right)$ found with RANSAC for each video frame t
- camera pose computed via non-linear optimization:

$$\min_{\mathbf{p}_t} \sum_j \left\| w_{tj} \left(\mathbf{x}_{tj} - \tilde{\mathbf{x}}_{tj} \right) \right\|^2 + \alpha^2 \left\| W \left(\mathbf{p}_t - \mathbf{p}_{t-1} \right) \right\|^2$$

- we regularize the solution to reduce virtual jitter
- α iteratively adjusted for each video frame:

$$\alpha^2 \left\| W \left(\mathbf{p}_t - \mathbf{p}_{t-1} \right) \right\|^2 \leq \sigma^2 N \quad \Rightarrow \quad \alpha^2 = \frac{\sigma^2 N}{\left\| W \left(\mathbf{p}_t - \mathbf{p}_{t-1} \right) \right\|^2}$$

video examples

in the future...

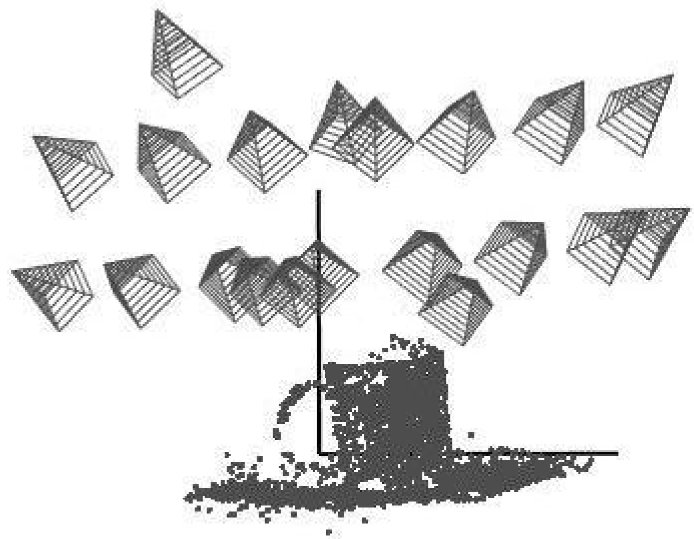
- optimize online computations for real-time performance:
  - SIFT recognition with a frame-to-frame feature tracker
- introduce multiple feature types:
  - SIFT features with edge-based image descriptors
- perform further testing:
  - scalability to large environments
  - multiple objects: real and virtual

questions?

http://www.cs.ubc.ca/~skrypnyk/arproject/

modelling reality: an example

20 input images



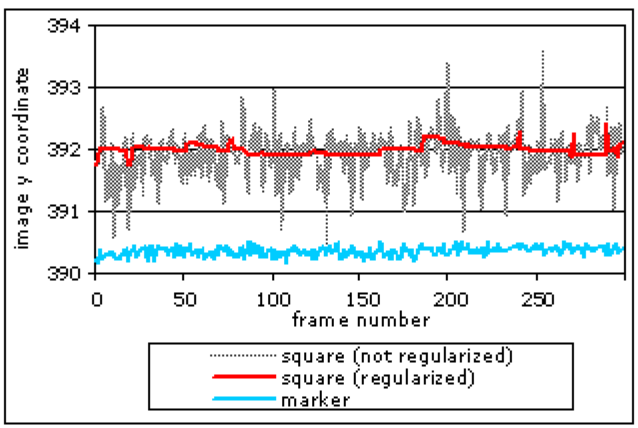
20 iterations: error = 0.2 pixels

registration accuracy

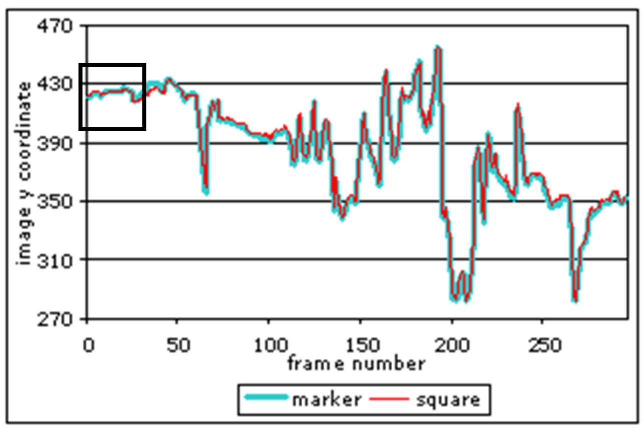
ground truth: ARToolKit marker

measurement: virtual square

stationary camera



moving camera



moving camera