Differentiable RANSAC for Camera Localization

Motivation
RANSAC (Random Sample Consensus) is an important algorithm in robust optimization. It cannot be used in deep learning pipelines because it is non-differentiable.

Our Goal: Use RANSAC in end-to-end learning.

RANSAC in a Nutshell:
1) Sample multiple hypotheses \( h \).
2) Score each hypothesis: \( s(h) \).
3) Take best one (and refine): \( h_{\text{best}} \).

Not differentiable

Contributions
- We explore two ways of making RANSAC differentiable: soft \( \arg\max \) selection and probabilistic selection.
- We put both options in a new end-to-end trainable camera localization pipeline.
- We show experimentally that probabilistic selection is superior. We call RANSAC with this option DSAC.
- We exceed state-of-the-art on camera localization by 7.2% (new results: by 12.8%).

Differentiable Alternatives to \( \arg\max \)
Our hypotheses depend on learnable parameters \( w \): \( h^w := h(w) \).

We wish to minimize the loss \( \ell \) of selected hypotheses over training images \( i \): \[ \hat{w} = \arg\min_w \sum_i \ell(h^w(i)) \]

We define a softmax score distribution:
\[ P(j|i) = \frac{\exp \left( f(h^w(i)) \right)}{\sum_k \exp \left( f(h^k(i)) \right)} \]

Soft \( \arg\max \) Selection
\[ h_{\text{SoftAM}} = \sum_k P(j|i) h^k \]

Hard, probabilistic decision

Derivatives
Soft \( \arg\max \) Selection:
\[ \frac{\partial}{\partial w} h_{\text{SoftAM}} = \sum_j \left( h^w \frac{\partial}{\partial w} P(j|i) + P(j|i) \frac{\partial}{\partial w} h^w \right) \]

Probabilistic Selection:
We derive the expectation of the task loss.
\[ \frac{\partial}{\partial w} E_j P(j|i) \left[ f(h^w) \right] = E_j P(j|i) \left[ \frac{\partial}{\partial w} h^w \right] \log P(j|i) + \frac{\partial}{\partial w} \left( \frac{f(h^w)}{\prod_k \exp \left( f(h^k) \right)} \right) \]

Application: Camera Localization
Training Data: Annotated Images
Test Input: RGB Image
Test Output: \( h \): 6D Pose
3D Translation \( t \)
3D Rotation \( \theta \)

Previous Work: Scene Coordinate Regression [Sho13, Bra16]

Scene Coordinate Regression
RANSAC + Refinement

Our Pipeline
Input RGB
Scene Coordinate (y)
Regression
Hypothesis Sampling
Scoring (s)
Hypothesis Selection
Refinement (R)

We learn two CNNs with parameters \( w \) and \( v \) jointly by minimizing:
\[ E_j P(j|i) \left[ f(R(h^w, v)^w, h^v) \right], \text{with} \quad f(h^w, h^v) = \max \{ f(\theta, \theta^v), \| t - t^v \| \} \]

New Results
7-Scenes: 68.0%
Median Accuracy on a Large Outdoor Scene:
PoseNet [Ken15]: 88cm, 1.0°
Sparse Features [Sat16]: 42cm, 0.5°
Our Results: 25cm, 0.5°

Experiments
Dataset: 7-Scenes [Sho13]
RGB-D images of 7 indoor scenes with pose annotations.
1k-5k training resp. test images.
We use only RGB at test time.

Initialisation: Componentwise Training
Scene Coordinate Regression:
Generate ground truth scene coordinates \( y \) and minimize:
\[ \hat{w} = \arg\min_w \sum_i \| y_i - y_i \| \]

Score Regression: Optimize a dummy score of hypotheses sampled around ground truth poses.

Results:
\[ \begin{array}{ccc}
\text{Scene} & \text{Componentwise} & \text{End-to-End} \\
\text{Input} & \text{Training} & \text{Training} \\
\text{w.r.t.} & \text{w.r.t.} & \text{End-to-End} \\
\text{Shotton et al. [Sho13]} & 38.0% & - \\
\text{Brachmann et al. [Bra16]} & 55.2% & - \\
\text{Ours, RANSAC} & 61.0% & - \\
\text{Ours, SoftAM} & 61.6% & 59.6% (2.0%) \\
\text{Ours, DSAC} & 60.3% & 62.4% (2.1%) \\
\end{array} \]

Componentwise Results:

Uncertainty

Scores 

Efficient & Effective Prioritized Matching for Large Images in Real-Time

Microsoft

© 2017 Microsoft Corporation. All rights reserved.

© 2017 Microsoft Corporation. All rights reserved.