Improved Geometric Verification for Large Scale Landmark Image Collections

Rahul Raguram, Joseph Tighe, and Jan-Michael Fram

Department of Computer Science
University of North Carolina at Chapel Hill


















Images of Rome

Photo Collection Modeling

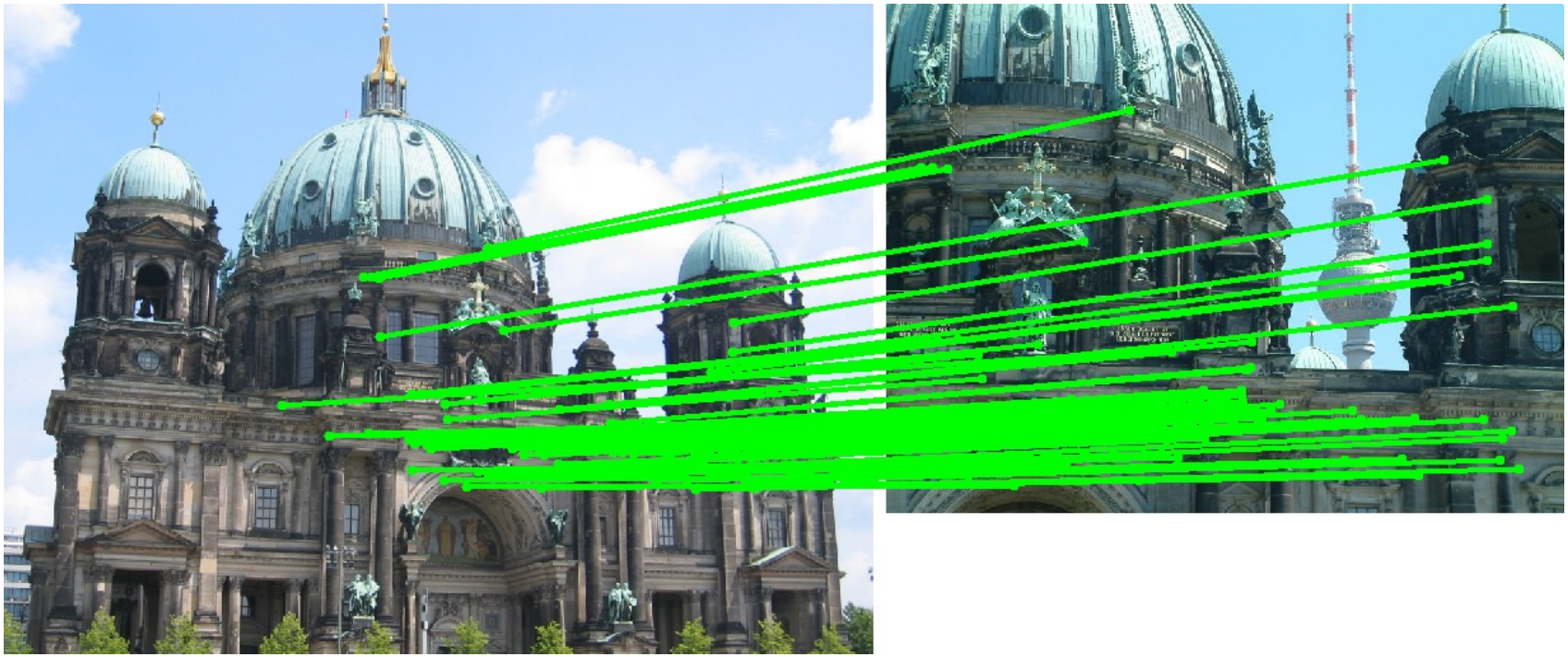
Challenges for Reconstruction

- Scalability/Speed
- Variety of cameras
- Appearance variability
- Resolution variability
- Model completeness
- Location completeness

None nearly addresses all challenges!

Geometric verification

- For a pair of images:
  - Detect feature points
  - Match features
  - Run RANSAC to find inliers

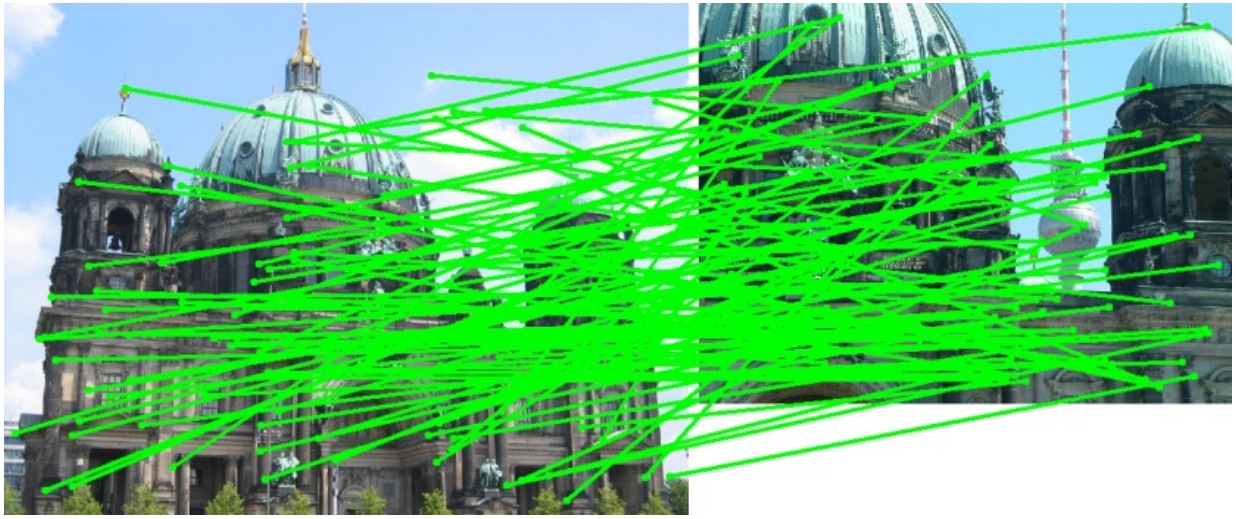


RANSAC Advances

- Many improvements to RANSAC
  - Preemptive RANSAC (Nister)
    - Breadth first search
  - WALDSAC (Chum and Matas)
    - Sequential decision theory
  - Real-time ARRSAC (Raguram et al.)
    - Combines preemptive RANSAC and WALDSAC
  - PROSAC (Chum and Matas)
    - Priority ordering

PROSAC

- Exploit image-to-image matching scores



1. Order matches based on matching scores
2. Sample from a small subset of the best points, gradually extend the size of the sampling pool

Motivation

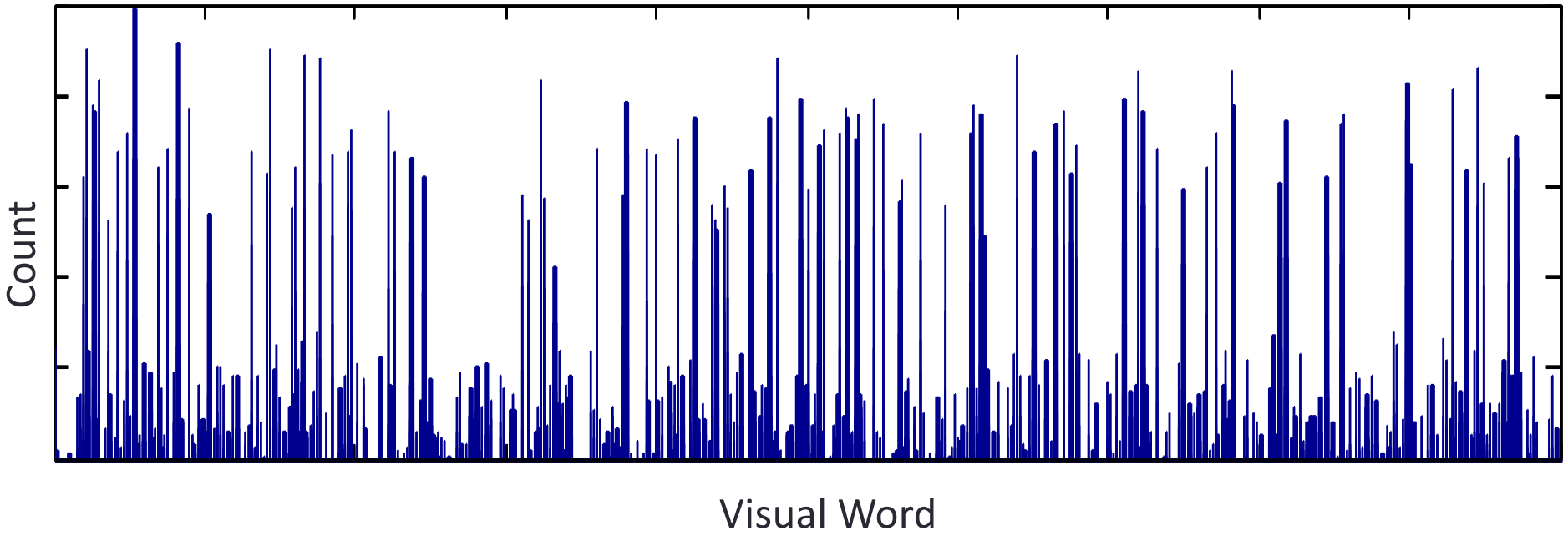
- Thus far, only image-to-image matching information has been considered
- PROSAC only requires a priority ordering on matches

- Key Insight: We can learn a better ordering metric from previous matches.

Adaptive Feature Prioritization

- Consider a visual vocabulary $W = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_N\}$
  - Obtained by running k-means on a random subset of points
- Carry out standard geometric verification:
  - For a pair of images: match features, run ARRSAC
  - If verified: update counts c_i for each visual word $\mathbf{w}_i$ that was found to be an inlier
- Idea: counts $C = \{c_i\}$ can be used as a learned priority order for PROSAC like sampling
  - Set the sample priority order to the ordering from these counts

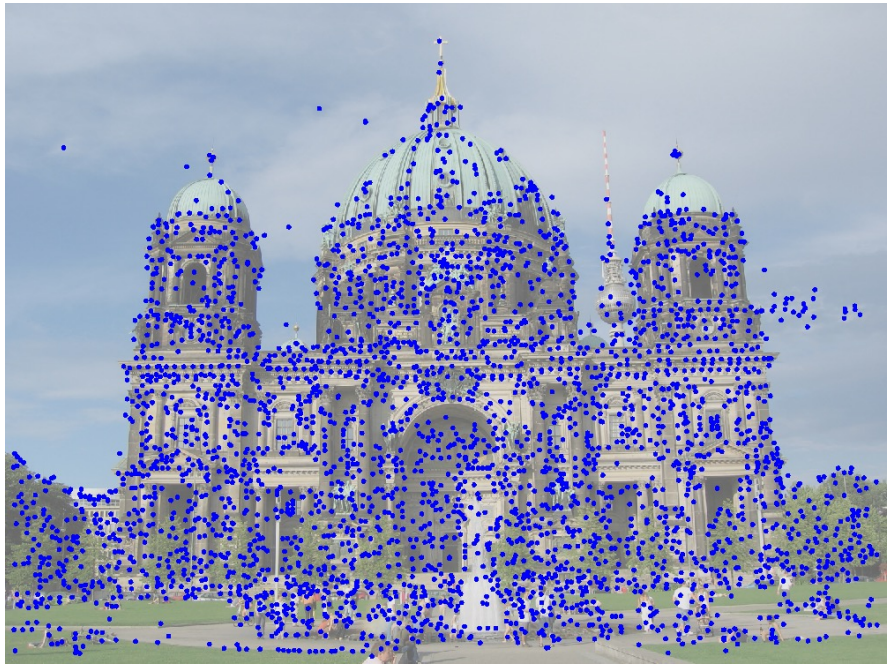
Adaptive Feature Prioritization



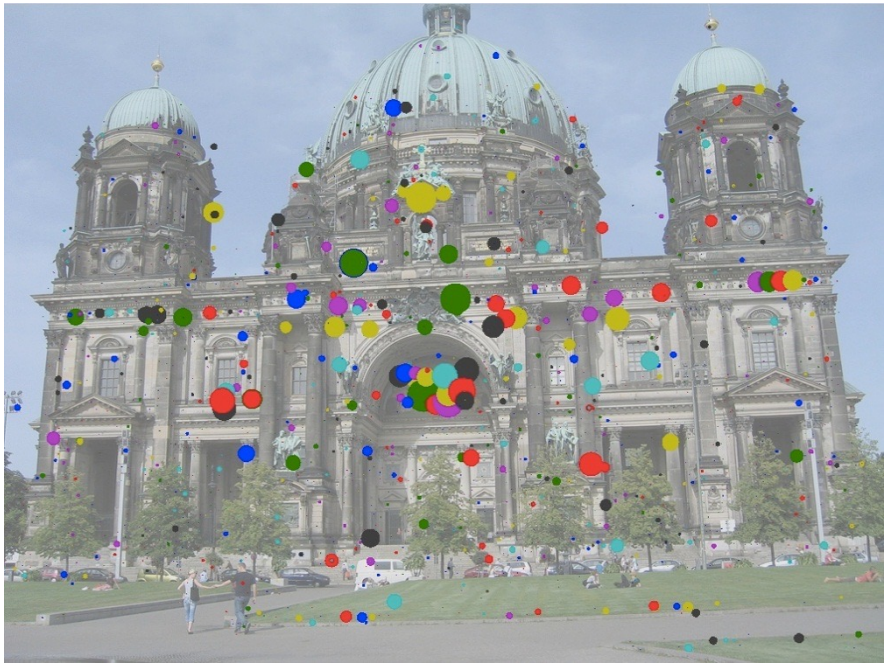
Evolving single feature inlier counts

Experiments: Berlin Dome (10k)

- Dataset:
  - 10k images for the Flickr query ‘Berlin Dome’
  - Relatively clean, with most images showing the Dome
- Randomly choose *k* image pairs and carry out geometric verification
  - Accumulate counts for visual vocabulary (20k words)



All detected sift features



Filtered features based on visual word counts ($c_i > 10$)

Experiments: Berlin Dome (10k)

- Experiment: Compare
  1. R1: baseline RANSAC
  2. R2: ARRSAC with ordering based on SIFT matching scores
  3. R3: ARRSAC with proposed ordering

	R1	R2	R3
Mean time per image pair	389.6 ms	41.2 ms	28.9 ms
Mean # of hypotheses per image pair	14218.4	604.1	399.7
Total runtime (hours:minutes)	23:48	04:08	03:16

- Observation: Sorting visual words based on results of previously rounds of geometric verification reveals useful information about the validity of feature matches

Experiments: Berlin (2.77M)

- Dataset:
  - 2.77 Million images for the Flickr queries
  - Very noisy
- Use “Building Rome on a Cloudless Day” (Frahm et al.):
  - Cluster images into 100,000 clusters
  - Perform geometric verification on each cluster independently
  - Clusters with 3 or more verified images are kept as iconic
    - All unverified images are discarded
- Accumulate counts for visual vocabulary (1 million words)

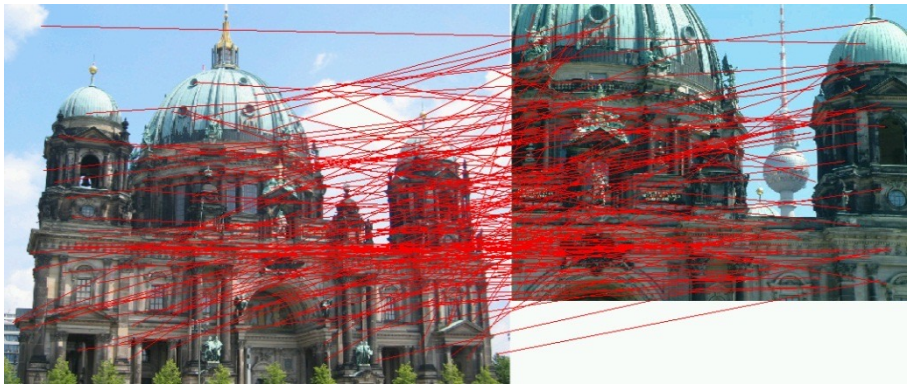
Experiments: Berlin (2.77M)

- Experiment: Compare
  1. R2: ARRSAC with ordering based on SIFT matching scores
  2. R3: ARRSAC with proposed ordering

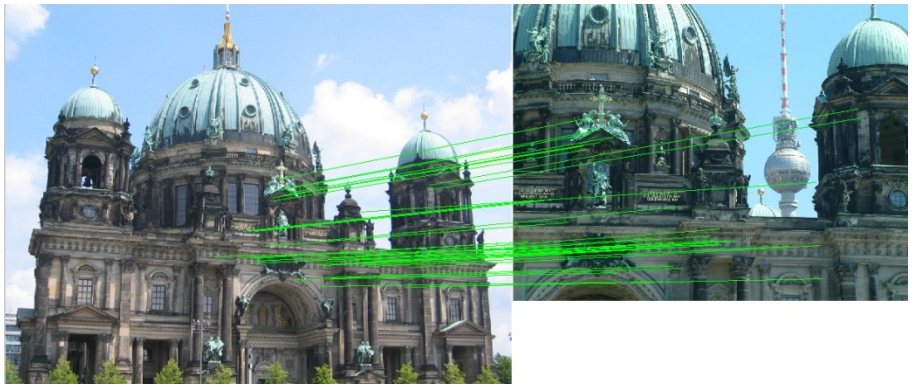
	R2	R3
Mean time per image pair	26.7 ms	18.8 ms
Total runtime (hours:minutes)	02:50	02:28
# of Iconics	9841	9912
# of Registered Images	132,719	133,830

- Only 5% of the entire dataset has been registered

Experiments: Berlin (2.77M)



Putative matches



Inliers (ε ≈ 0.20)

	RANSAC	ARRSAC w/ match scores	ARRSAC w/ inlier counts
# of hypotheses	1.07×10^6	22000	595
Run time	2.77×10^4 ms	818.3 ms	27.5 ms

Addressing Completeness

- We wish to register the 2.64 million image that were not registered in the first step
- For each rejected image:
  - Retrieve the 20 most visually similar iconic images
  - Perform geometric verification
- This process alone takes 1 day, 10 hours. This is too slow.

Key Insight: We can learn what a “good” image looks like.

Addressing Completeness

- Train a “landmark” image classifier from the iconic images already found.
  - Positive examples: the iconic image from each cluster (9,471 images)
  - Negative examples: random set of rejected images (10,529 images)
- Use our dictionary to compute a bag-of-visual-words histogram
  - While this feature has 1 million dimensions, note it is very sparse with on average only 2,000 words
- Train a linear SVM

Positive Training Images (”Iconic” Images)

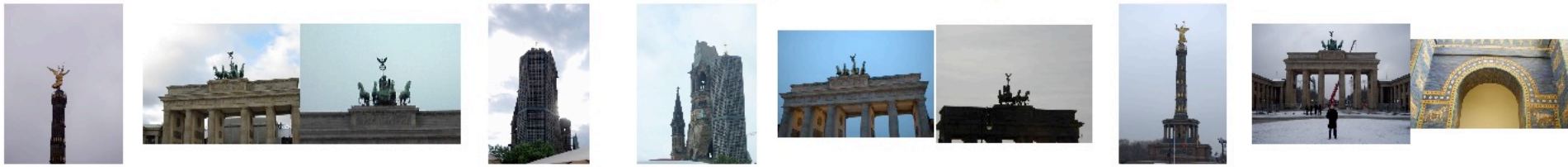
Negative Training Images (Rejected Images)

Experiments: Berlin (2.77M)

- Verify only rejected images that are classified as “landmarks”
  - 26% of rejected images are classified as “landmarks”

	Full re-verification	SVM+re-verification
# of newly registered images	273,639	202,538
Total runtime (days:hours:minutes)	01:10:26	00:10:43

Highest Scoring Verified Images



2.25 2.21 2.16 2.04 1.84 1.70 1.68 1.62 1.61 1.60

Lowest Scoring Verified Images

-1.39 -0.81 -0.74 -0.62 -0.59 -0.58 -0.58 -0.57 -0.57 -0.56

Summary

- Combining recognition and geometry methodologies enables fast, more complete registration.
- Learned priors based on match history:
  - Captures useful information about the low-level features which are useful for the specific task of interest (geometric verification)
- Using a fast pre-classification step can greatly decrease runtime while minimally impacting matching performance.