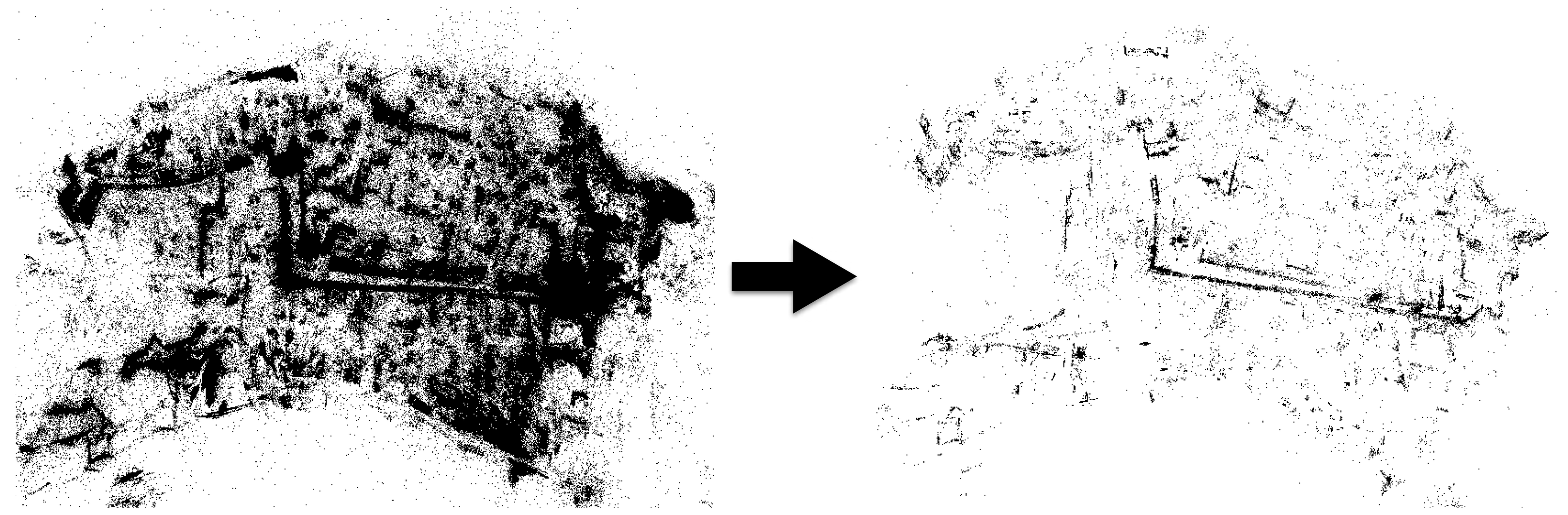
Minimal Scene Descriptions from Structure from Motion Models

Song Cao and Noah Snavely

Department of Computer Science, Cornell University

Introduction

- **How much data do we need to describe a location?**
- Context: 3D scene reconstructions by Structure from Motion
- Goal: Compute **compact** representations of SfM reconstructions for location recognition
- Benefits: Reduce the memory and computational cost of a location recognition system
- Take-home message: We can summarize an SfM model with $<2\%$ of points, while keeping reasonable recognition performance, aided by selecting distinctive points.



Original Points
# of points: 1,886,884
Registration performance: 99.50%

Selected Points
# of points: 31,752
Registration performance: 93.38%

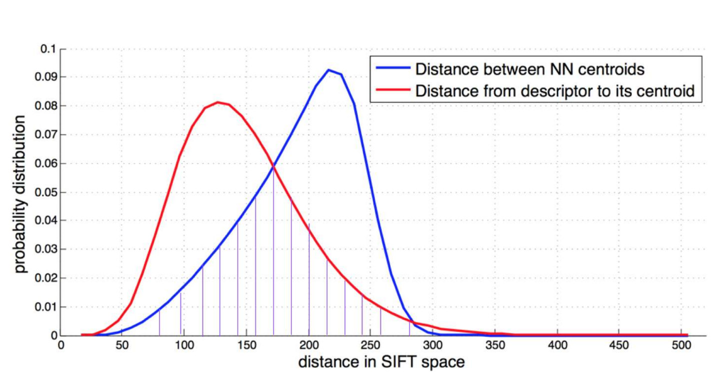
Input from Structure from Motion

- An image set $\mathcal{I}$ of size m and 3D point set $\mathcal{P}$ of size n ($n \gg m$)
- Visibility matrix M of size $m \times n$: $M_{ij} = \begin{cases} 1, \text{point } P_j \text{ is visible in image } I_i \\ 0, \textit{otherwise} \end{cases}$
- A descriptor mean for each 3D point

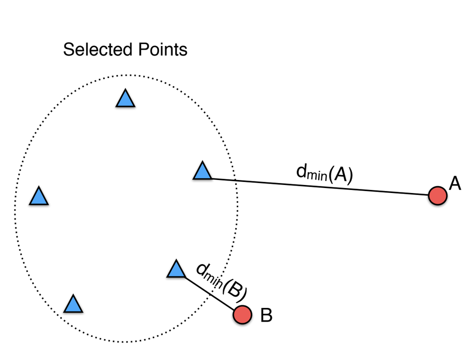
Objectives

- Goal: Compute a small subset $\mathcal{P}'$ of $\mathcal{P}$ that captures as much data as possible
- Previous Approach [1]: K-cover algorithm - greedy algorithm that maximizes **coverage**
- Our Approach: an point selection algorithm that considers
  - **1. coverage:** any new image has a high probability of seeing a large number of points in $\mathcal{P}'$
  - **2. distinctiveness:** the descriptors in $\mathcal{P}'$ are sufficiently distinct from one another

Why Distinctiveness?



- Large portion of descriptors are confusing!
- Select points that both ensure **coverage** and **distinct** reduces errors in matching process

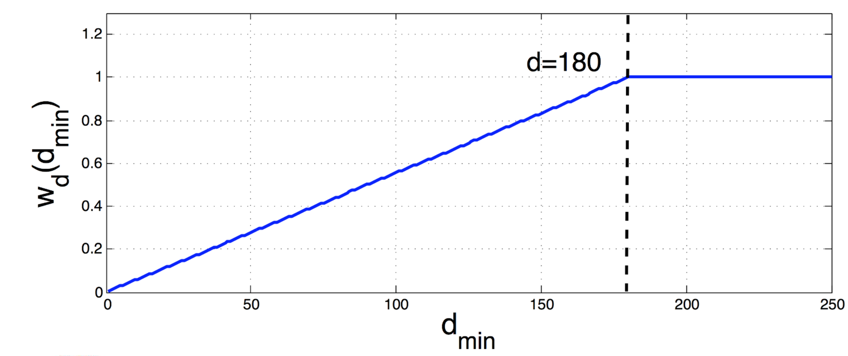


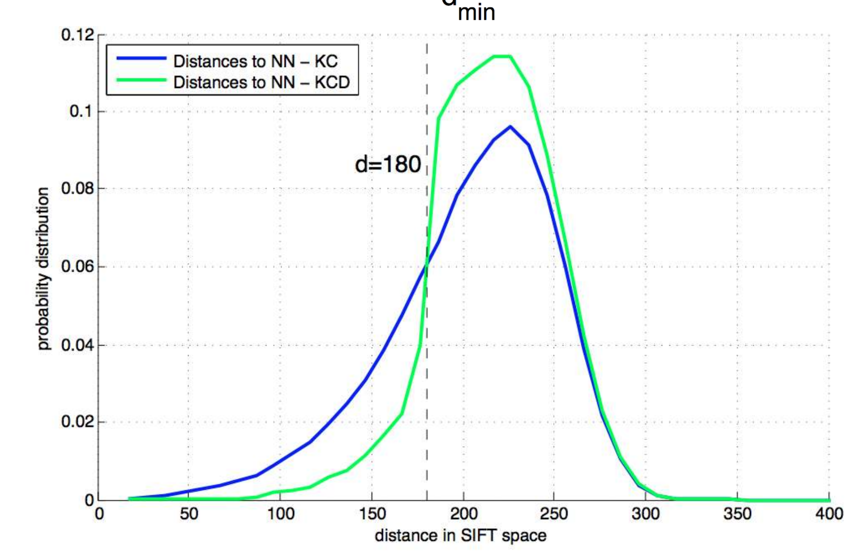
Maximizing Expected Coverage

- Treat visibility as probabilistic event: P_j is visible in each database image I_i with probability p_{ij}
- Goal: to find a subset $\mathcal{P}'$ that maximizes the probabilities of each image seeing $\geq K$ points in $\mathcal{P}'$
$$S(\mathcal{P}') = \sum_{i \in \mathcal{I}} \Pr(v_{i,\mathcal{P}'} \geq K)$$
- Gain of adding point P_j: $G(j, \mathcal{P}') = S(\mathcal{P}' \cup \{P_j\}) - S(\mathcal{P}')$
- Bootstrapping problem: If image I_i sees fewer than $K-1$ points in $\mathcal{P}'$, then the gain for adding any new point to $\mathcal{P}'$ w.r.t. I_i is zero
- Initial point set: We first need to cover each image with K points to yield a non-zero gain

Selecting an Initial Set of Distinctive Points

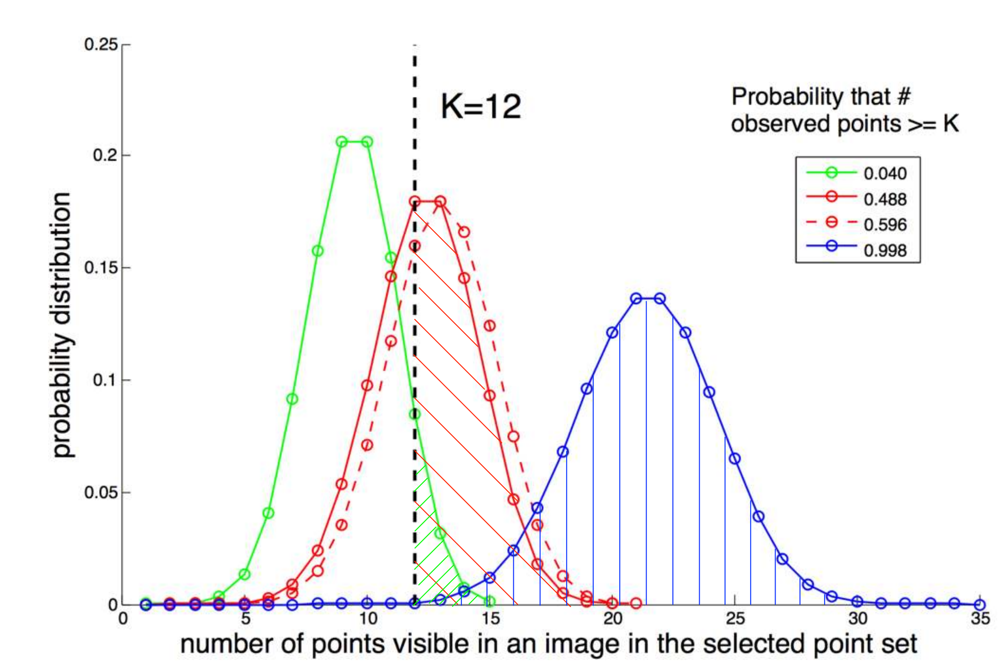
1. Gain of adding point P_j by K-cover (KC) algorithm [1]
$$G_{KC}(j, \mathcal{P}') = \sum_{I_i \in \mathcal{I} \setminus C} M_{ij}$$
2. Weight factor for encouraging **distinctiveness** ($d_{\min}(j)$ is the nearest distance from P_j to current selected $\mathcal{P}'$)
$$w_d(d_{\min}(j)) = \begin{cases} d_{\min}(j)/d, & d_{\min}(j) < d \\ 1, & d_{\min}(j) \geq d \end{cases}$$
3. Greedily select the point with highest weighted gain
$$G_{KCD}(j, \mathcal{P}') = w_d(d_{\min}(j)) G_{KC}(j, \mathcal{P}')$$
4. Repeat Step 3 until all images are covered by at least K points





Probabilistic K-cover Algorithm

1. Assuming constant p for each p_{ij}, the number of points in the chosen subset $\mathcal{P}'$ image I_i sees follows binomial distribution
$$\Pr(v_{i,\mathcal{P}'} = K') = \binom{C_i}{K'} p^{K'} (1-p)^{C_i - K'}$$
2. Gain of adding point P_j (e.g. dotted red v.s. red on the right)
$$G_{KCP}(j, \mathcal{P}') = \sum_{i \in \mathcal{I} \setminus C} p_{ij} \Pr(v_{i,\mathcal{P}'} = K - 1)$$
3. Greedily choose the point P_{j^*} that maximizes $G_{KCP}(j, \mathcal{P}')$ and update $\Pr(v_{i,\mathcal{P}'} = K')$
4. Repeat from Step 3 until a specified percentage of images are covered.



- Distributions of three images covered by 15, 20 and 35 points respectively

Datasets

Dataset	# DB Imgs	# 3D Points	# Queries
Dubrovnik [1]	6,044	1,886,884	800
Aachen [2]	4,479	1,980,036	369
Landmarks [3]	205,813	38,190,865	10,000

Registration Performance

- Methods: the K-cover algorithm (KC)[1], our initial point set selection algorithm only (KCD), and our full approach including the probabilistic K-cover algorithm (KCP)
- Compare the performances of scene descriptions with the same number of points

Dubrovnik Dataset [1]
query images: 800, registered by full set: 99.50%

K	12 (9)	20 (12)	30 (20)	50 (35)
# points	5,788	10,349	17,241	31,752
% points	0.31%	0.55%	0.91%	1.68%
KC	58.00%	77.06%	86.00%	91.81%
KCD	62.88%	78.88%	**87.38%**	92.50%
KCP	**64.25%**	**79.13%**	87.25%	**93.38%**

Aachen Dataset [2]
query images: 369, registered by full set: 88.08%

K	30 (20)	50 (32)	80 (52)	100 (65)
# points	13,299	23,675	40,377	52,161
% points	0.67%	1.20%	2.04%	2.63%
KC	50.95%	62.06%	66.40%	71.27%
KCD	54.20%	63.14%	69.38%	72.36%
KCP	**56.37%**	**64.23%**	**70.19%**	**73.98%**

Landmarks Dataset [3]
query images: 10,000, registered by full set: 94.33%

K	6 (4)	9 (6)	12 (9)	20 (12)
# points	140,306	222,161	311,035	571,864
% points	0.37%	0.58%	0.81%	1.50%
KC	44.84%	59.86%	69.56%	81.06%
KCD	45.45%	61.26%	70.59%	81.04%
KCP	**45.90%**	**61.50%**	**71.87%**	**81.45%**

Reference

[1] Y. Li, N. Snavely, and D. Huttenlocher. Location recognition using prioritized feature matching. In ECCV, 2010.

[2] T. Sattler, T. Weyand, B. Leibe, and L. Kobbelt. Image retrieval for image-based localization revisited. In BMVC, 2012.

[3] Y. Li, N. Snavely, D. Huttenlocher, and P. Fua. Worldwide pose estimation using 3d point clouds. In ECCV, 2012

Department of Computer Science, Cornell University, Ithaca, NY, USA

Email: caosong@cs.cornell.edu

WWW: http://www.cs.cornell.edu/projects/minimalscene/