RANSAC: RANdom Sampling And Consensus

Roland Angst

rangst@stanford.edu

www.stanford.edu/~rangst

The Need for RANSAC

- Why do I need RANSAC? I know robust statistics!
  - “Robust Statistics” Huber [1981]
    - M-estimator, L-estimator, R-estimators, …
  - Least Median of Squares (LMedS), …
- Breakdown point of an estimator
  - “Proportion of incorrect observations … an estimator can handle before giving an incorrect … result” [Wikipedia]
- Robust estimators can achieve breakdown point of 50%
  - For example: median
- Usually a non-linear, non-convex optimization problem needs to be solved

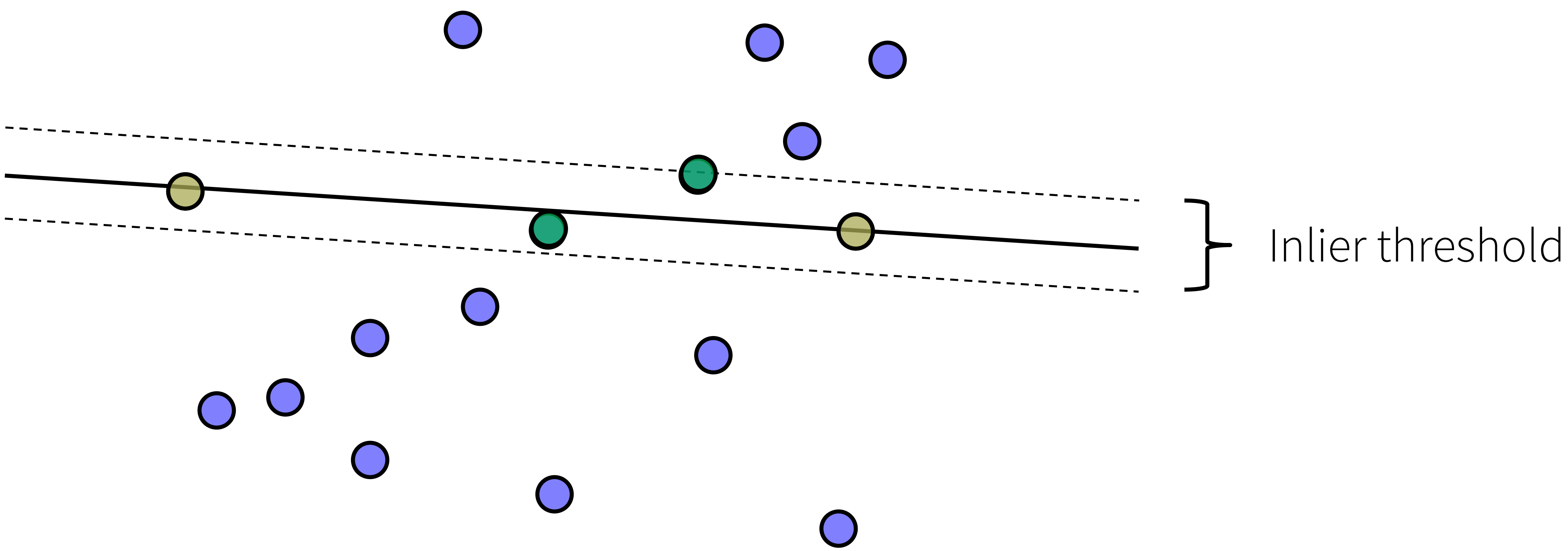
The Need for RANSAC

- Problems
  - Estimators for more complex entities (eg. homographies, essential matrices, ...)?
  - Inlier ratio of computer vision data can be lower than 50%
- Hough Transform
  - Excellent candidate for handling high-outlier regimes
  - Can only handle models with very few parameters (roughly 3)
- RANSAC is a good solution for models with slightly larger number of parameters
  - Roughly up to 10 parameters (depending on inlier ratio)

RANSAC [Fischler & Bolles 81]

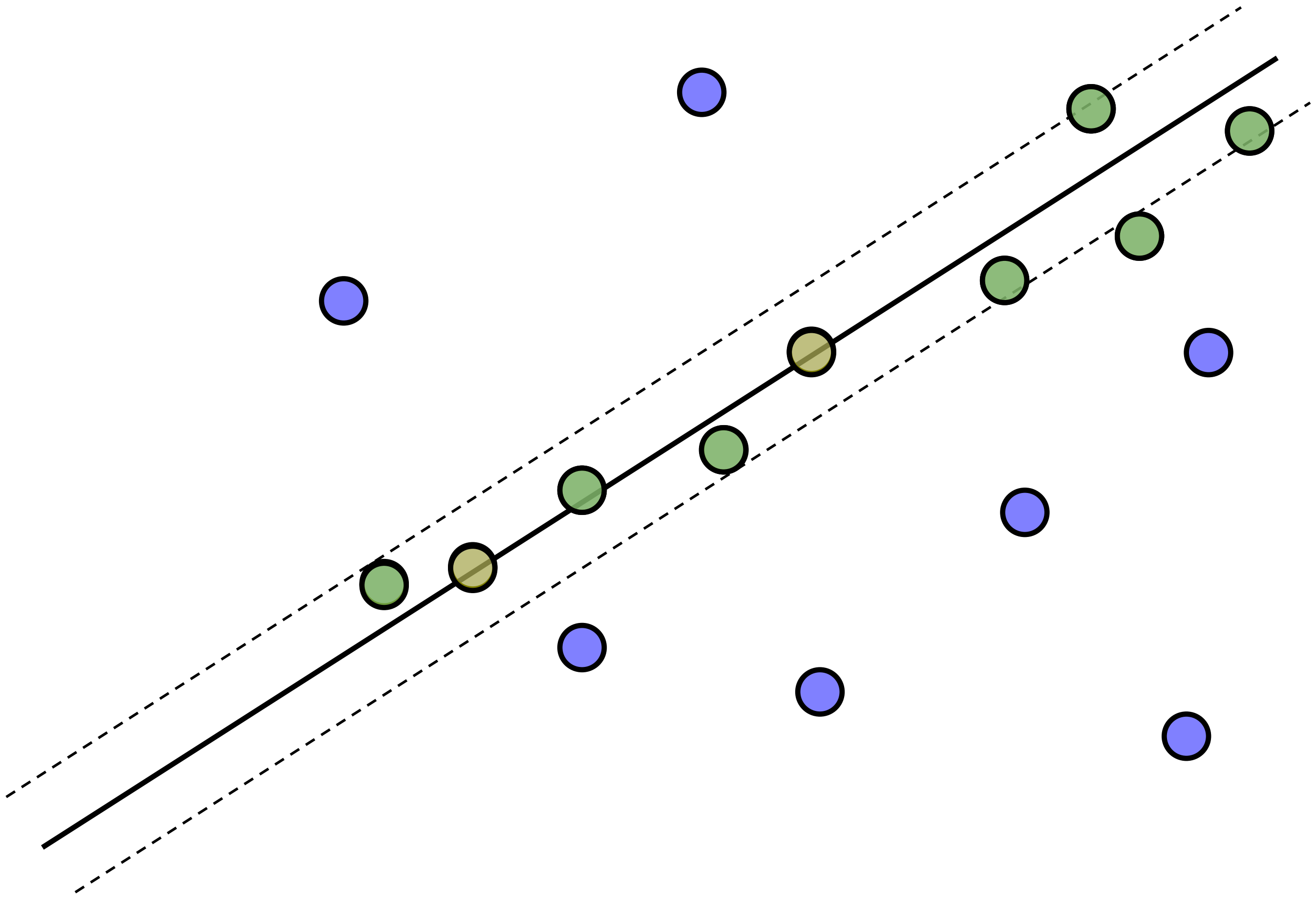
- Hypothesize-and-verify framework
  - Sample hypothesis and verify with data
- Assumptions
  - Outliers provide inconsistent (ie. random) votes for models
  - There are sufficiently many inliers to detect a correct model
- Hypothesis generation
  - Sample subset of data points and fit model parameters to this subset
    - Plain RANSAC: sample points uniformly at random
- Verification on all remaining data points

Algorithm Outline



1. Select random sample of minimum required size to fit model parameters
2. Compute a putative model from sample set
3. Verification stage: Compute the set of inliers to this model from whole data set
4. Check if current hypothesis is better than any other of the previously verified
5. Repeat 1-4

Algorithm Outline



1. Select random sample of minimum required size to fit model parameters
2. Compute a putative model from sample set
3. Verification stage: Compute the set of inliers to this model from whole data set
4. Check if current hypothesis is better than any other of the previously verified
5. Repeat 1-4

Number of Iterations

- Probability of selecting an inlier given by inlier ratio p_{inlier}
- Sample size s
- Confidence value for having sampled at least one all-inlier sample P
- Number of iterations k
- Let's put all of this together: $1 - P \geq (1 - p^s_{\text{inlier}})^k$

Probability of having selected at least one outlier in each of the k trials

P = 0.99; proportion of outliers							
s	5%	10%	20%	25%	30%	40%	50%
2	2	3	5	6	7	11	17
3	3	4	7	9	11	19	35
4	3	5	9	13	17	34	72
5	4	6	12	17	26	57	146
6	4	7	16	24	37	97	293
7	4	8	20	33	54	163	588
8	5	9	26	44	78	272	1177

$$k \geq \frac{\log(1 - P)}{\log(1 - p^s_{\text{inlier}})}$$

RANSAC Parameters

- How to find inlier ratio?
  - Provide lower bound for initialization and recompute when new best hypothesis has been found
- Scale of inlier noise
- Confidence for having sampled at least one all-inlier sample

Shortcomings of ‘Plain’ RANSAC

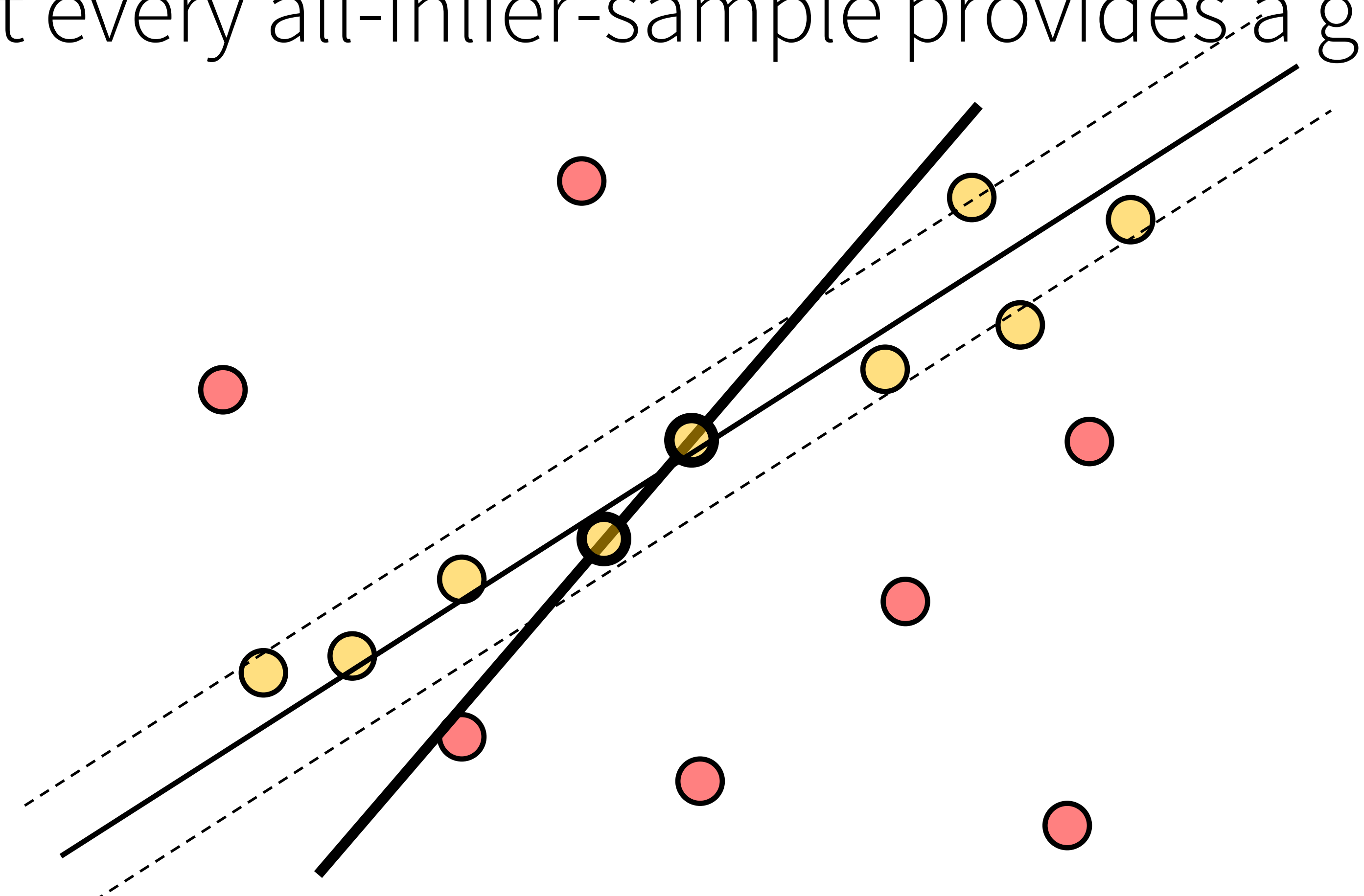
- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Shortcomings of ‘Plain’ RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Noisy Inliers

- Problem: not every all-inlier-sample provides a good solution



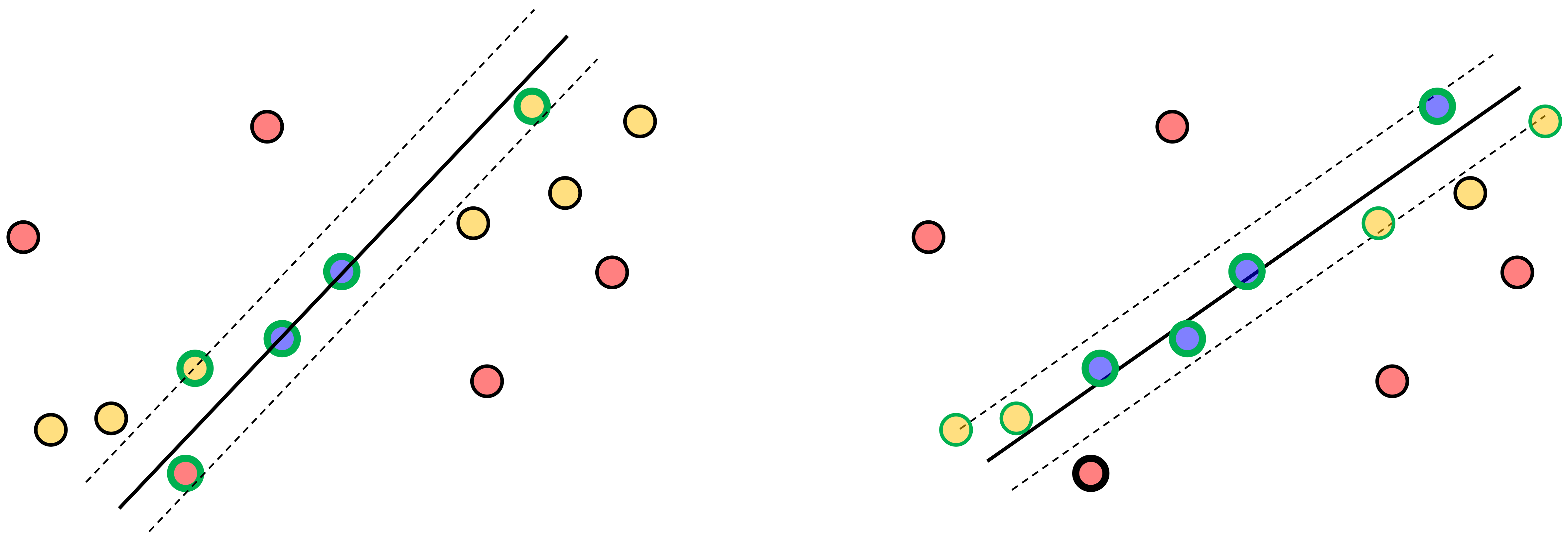
  - Sampling more than one all-inlier-set might be necessary!
  - In practice, solution often found only after roughly $k = \left(\frac{1}{p_{\text{inlier}}}\right)^{\text{sample size}}$ iterations
    - Simple calculation $k \geq \frac{\log(1-P)}{\log(1-p_{\text{inlier}}^{s})}$ is inaccurate

Shortcomings of ‘Plain’ RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Increase Accuracy of Estimated Models

- Lo-RANSAC
  - Run inner RANSAC loop with non-minimal sample size to refine hypothesis of minimal sample size
  - “Locally Optimized RANSAC “ Chum, Matas, Kittler [DAGM03]



- MLESAC
  - Fit model by max likelihood rather than max inlier count
  - “MLESAC: A new robust estimator with application to estimating image geometry” Torr & Zisserman [1996]

Shortcomings of ‘Plain’ RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Handling Degenerate Cases

- “Two-view geometry estimation unaffected by a dominant plane” Chum et.al. [CVPR05]
  - Estimate fundamental
  - If successful try to fit homography to **triplet** of 7-cardinalty MSS
    - If homography can be found run plane-and-parallax fundamental estimation
      - 2 points off the plane need to get fundamental from known homography
      - 2-pt RANSAC over outliers of homography
    - else non-planar case

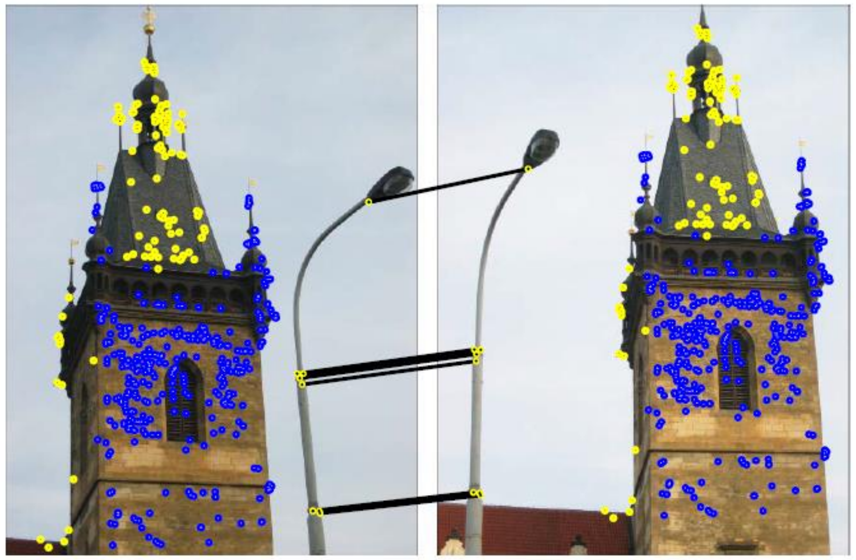


Figure 1: The LAMPPOST scene with 97% of correct tentative correspondences lying in or near a dominant plane. In 100 runs, RANSAC fails to find a single inlier on the lamp 83 times; in the remaining 17, no more than 4 out of the 10 correspondences on the lamppost are found. Points on the lamppost are far from the dominant plane and therefore critically influence the precision of epipolar geometry and egomotion estimation. The DEGENSAC algorithm, with the same computational complexity as RANSAC, found the 10 lamppost inliers in all runs. Corresponding points lying in the dominant plane are dark, off-the-plane points are light, and the points on the lamp are highlighted by line segments.

- Other approaches for making RANSAC robust w.r.t. degeneracies
  - “RANSAC for (quasi-)degenerate data (QDEGSAC)” Frahm & Pollefeys [CVPR06]

Shortcomings of 'Plain' RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Hypothesis Generation

- Trade-off between exploration and exploitation
  - Previously verified hypothesis tell us something about inlier set
  - Still, we should avoid narrowing our search too quickly
- Especially important for multi-model case
  - Eg. estimation of multiple planes in a scene
  - Points on other planes act as outliers to plane under consideration

PROSAC

- "Matching with PROSAC – progressive sample consensus" Chum & Matas [CVPR05]
- Use of a-priori knowledge
  - Confidence of a matching pair (eg. based on descriptor matching distance)
- PROSAC: Favor high-quality matches while sampling points for minimal sample
  - Sort correspondences according to matching score
  - Consider progressively larger subsets of putative correspondences
  - Note: draws the **same** samples as RANSAC would, just in different order
- Pro
  - Can decrease the number of required hypothesis considerably
- Contra
  - Performance gain depends on data
  - Practical observation: high-confidence matches appear often appear in clusters on same spatial structure
    - Degenerate configurations…

Shortcomings of ‘Plain’ RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered in remainder

Verification

- Phrase hypothesis verification in sequential testing framework
  - Subsample remaining data and verify on this subset
  - If inlier ratio is sufficiently low: terminate verification
- Several papers have been published
  - Threshold determined based on Td,d tests
    - "Randomized RANSAC with Td,d test" Matas, Chum [IVC04]
  - Bail-Out test based on hyper-geometric distribution
    - "An effective bail-out test for RANSAC consensus scoring" Capel [BMVC05]
  - Wald's Sequential Probability Ratio Test (WaldSAC)
    - "Optimal randomized RANSAC" Chum & Matas [PAMI07]

Speedup of 2-7 times compared to standard RANSAC according to: "A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus" Raguram et.al. [ECCV08]

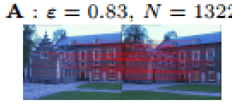
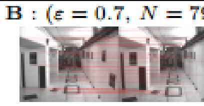
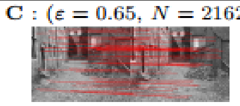
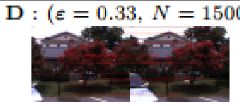
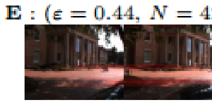
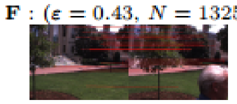
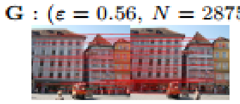
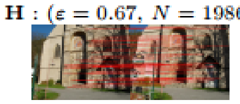
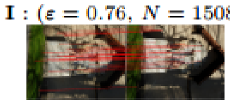
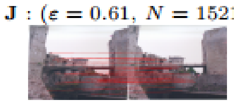
Preemptive RANSAC

- "Preemptive RANSAC for live structure and motion estimation" Nister [ICCV03]
- Find a good estimate within a fixed time budget (eg. in a vSLAM system)
- Idea
  - Generate fixed number of hypothesis
  - Verify all of them in parallel
    - Breadth-first verification scheme
    - Verify all hypothesis on a subset of the data
      - Prune unpromising hypothesis and retain promising ones
    - Verify on increasingly larger subsets, followed by pruning step

ARRSAC

- Adaptive Real-Time RANSAC
- Carefully designed combination of previous RANSAC approaches
- Achieves considerable speed-ups while still providing correct solution
- “A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus” Raguram et.al. [ECCV08]

Table 1. Evaluation results for ten selected real image pairs. The images show variation over a range of inlier ratios and number of correspondences. It can be observed from the above results that the ARRSAC approach produces significant computational speed-ups, while simultaneously providing accurate robust estimation in real-time. In practice, the ARRSAC technique stays well within the time budget, with estimation speeds ranging between 55-350 Hz. It can be seen from the table that the number of hypotheses evaluated by ARRSAC is always less than preemptive RANSAC. In addition, the correct epipolar geometry is always recovered.

		RANSAC	$T_{d,d}$	Bail-out	Wald	PROSAC	Pre-emptive	ARRSAC
A : $\varepsilon = 0.83$, $N = 1322$	I	884	884	885	889	885	933	1099
	k	112	156	118	148	5	500	6
	vpm	1322	531	361	594	1322	396	434
	spd-up	1.0	1.9	2.6	2.2	12.9	0.4	37.3
B : ($\varepsilon = 0.7$, $N = 795$)	I	514	512	514	517	514	527	559
	k	242	329	242	283	43	500	47
	vpm	795	149	65	132	795	374	218
	spd-up	1.0	2.4	4.1	3.9	7.7	0.3	11.1
C : ($\varepsilon = 0.65$, $N = 2162$)	I	1344	1342	1344	1345	1344	1372	1413
	k	208	947	208	211	17	500	17
	vpm	2162	59	67	52	2162	400	430
	spd-up	1.0	1.9	5.5	5.8	15.2	0.9	29.1
D : ($\varepsilon = 0.33$, $N = 1500$)	I	483	483	484	484	484	350	491
	k	15506	49090	15509	15514	201	500	241
	vpm	1500	22	37	28	1500	398	59
	spd-up	1.0	1.9	5.7	6.2	104.2	90.6	305.7
E : ($\varepsilon = 0.44$, $N = 420$)	I	174	174	173	177	177	134	183
	k	2714	4889	2719	2716	439	500	499
	vpm	420	10	24	13	420	305	70
	spd-up	1.0	2.0	4.4	5.7	5.3	11.5	20.2
F : ($\varepsilon = 0.43$, $N = 1325$)	I	557	557	557	557	557	488	582
	k	4353	11753	4366	4359	288	500	329
	vpm	1325	5	31	24	1325	396	182
	spd-up	1.0	2.1	4.7	5.3	6.9	12.3	15.1
G : ($\varepsilon = 0.56$, $N = 2875$)	I	1537	1524	1539	1537	1537	1539	1616
	k	432	628	437	432	30	500	33
	vpm	2875	20	24	31	2875	400	456
	spd-up	1.0	2.2	3.1	4.2	16.6	1.3	47.2
H : ($\varepsilon = 0.67$, $N = 1986$)	I	1249	1247	1251	1252	1250	1237	1330
	k	335	1153	351	347	17	500	18
	vpm	1986	29	41	32	1986	400	467
	spd-up	1.0	2.4	4.2	4.9	22.2	1.1	60.9
I : ($\varepsilon = 0.76$, $N = 1508$)	I	945	934	948	951	949	960	1149
	k	498	1644	505	502	58	500	65
	vpm	1508	43	31	33	1508	398	277
	spd-up	1.0	1.9	2.7	2.5	12.3	1.8	45.1
J : ($\varepsilon = 0.61$, $N = 1521$)	I	872	872	873	873	872	870	927
	k	362	1167	381	372	40	500	44
	vpm	1521	39	28	25	1521	398	338
	spd-up	1.0	2.2	6.1	6.3	9.8	1.9	26.1

Further Evaluation and Comparisons

- “Performance Evaluation of RANSAC Family” Choi et.al. [BMVC09]

Inlier Ratio	0.300	0.400	0.500	0.600	0.700	0.800	0.900
LMedS	**27.320**	**6.461**	1.321	1.356	1.408	1.379	1.509
RANSAC	1.305	1.323	1.326	1.330	1.390	1.415	1.483
MSAC	1.229	1.266	1.284	1.337	1.373	1.415	1.535
MLESAC	1.248	1.269	1.289	1.316	1.358	1.410	1.446
LO-RANSAC	1.245	1.229	1.229	1.221	1.229	1.253	1.255
R-RANSAC.T	1.317	1.323	1.304	1.341	1.394	1.401	1.475
R-RANSAC.S	**15.210**	1.848	1.131	1.229	1.291	1.370	1.389
FH' MAPSAC	**7.708**	**2.036**	1.647	1.484	1.463	1.490	1.526
AMLESAC	**2.051**	1.477	1.452	1.529	1.517	1.526	1.575
GASAC	**28.640**	**7.370**	1.108	1.077	1.100	1.120	1.147
pbM-estimator	1.023	1.034	1.209	1.255	1.286	1.291	1.355
uMLESAC	**5.246**	1.382	1.402	1.383	1.398	1.433	1.489
RANSAC*	**47.010**	**13.920**	**3.031**	1.688	1.386	1.255	1.187
MLESAC*	**50.110**	**10.240**	**2.839**	1.694	1.352	1.235	1.145

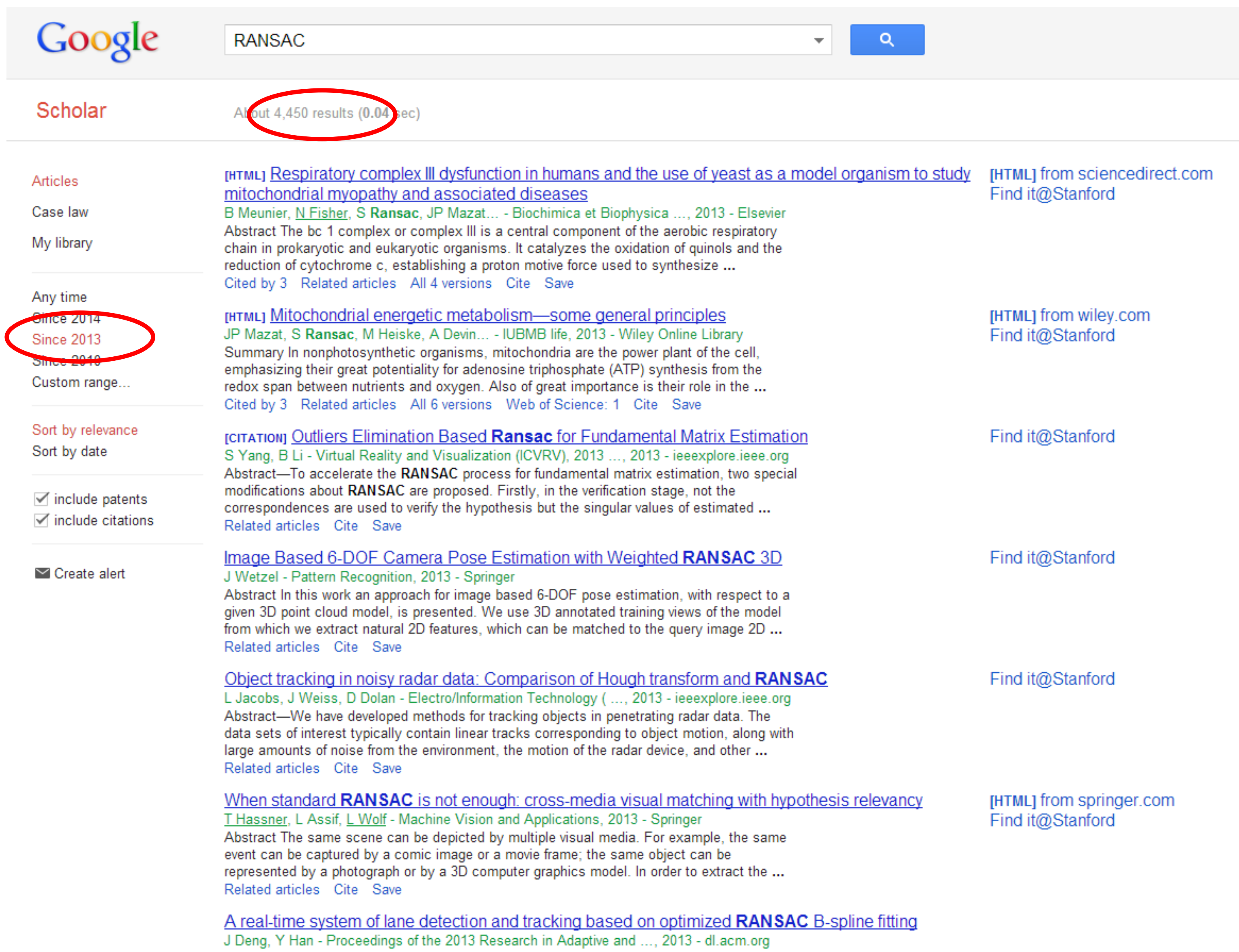
Mag. of Noise	0.250	0.500	1.000	2.000	4.000
LMedS	1.379	1.372	1.295	1.256	1.228
RANSAC	1.385	1.380	1.315	1.282	1.259
MSAC	1.389	1.319	1.306	1.246	1.193
MLESAC	1.364	1.341	1.309	1.262	1.218
LO-RANSAC	1.219	1.224	1.202	1.203	1.222
R-RANSAC.T	1.396	1.363	1.286	1.232	1.180
R-RANSAC.S	1.253	1.325	1.413	1.583	1.618
FH'MAPSAC	1.458	1.557	1.513	1.395	1.406
AMLESAC	1.479	1.499	1.942	1.192	1.164
GASAC	1.098	1.100	1.102	1.115	1.124
pbM-estimator	1.258	1.272	1.264	1.296	1.306
uMLESAC	1.361	1.232	1.069	1.025	1.051
RANSAC*	1.389	1.379	1.409	1.440	1.552
MLESAC*	1.357	1.326	1.330	1.388	1.468

Shortcomings of ‘Plain’ RANSAC

- Scale of inlier noise (for inlier-outlier threshold) needs to be specified
- Correct model is not generated with user-defined confidence
- Estimated model might be inaccurate
- Degenerate cases not handled
- Can be sped up considerably
  - Better hypothesis generation
  - Faster verification schemes
- Multiple models
  - Model selection
  - Interesting problem, but not covered here

RANSAC Conclusion

- Many different 'flavours'
- Still an active research area



?