Reduced Epipolar Cost for Accelerated Incremental SfM[1]

A.L. Rodríguez[2], P.E.López-de-Teruel[2], A.Ruiz[3] {*alrl1*, *pedroe*, *aruiz*}@*um.es*

Universidad De Murcia

PARP Research Group

CVPR 11 COLORADO

Problem:

Bundle Adjustment (BA) is usually the main performance bottleneck in incremental SfM.

Proposal:

Complement BA with a previous gobal epipolar adjustment (GEA): GEA optimization provides a map very close to the BA optimal configuration, requiring less BA iterations to reach the optimal.

Simplified algebraic epipolar cost has computational and qualitative advantages for fast/real-time performance. Prior data normalization approximates the algebraic cost to geometric; in practice algebraic cost offers a competitive trade-off between *performance time* and *error reduction*.

1. Global epipolar cost error

$$C_{ege} = \sum_{p} \sum_{i[p]} \sum_{j[p]} d(x_i^{(p)}, E_{ij}\, x_j^{(p)})$$

Where:

$$E_{ij} = R_i \left[\frac{T_j - T_i}{||T_j - T_i||} \right]_{\times} R_j^T$$

E_{ij} → Essential matrix for epipolar geometry between views *i* and *j*.

p → *2D feature tracking for 3D point.*

R_i, T_i → Rotation matrix and center vector for the *i*-th camera.

2. Global epipolar adjustment

SfM map optimization:

(1) Initialization: computation of new view-pair epipolar constraints.

(2) Levenberg-Marquart optimization: find optimal value for the epipolar cost C_{ege} (iterative, on cameras only).

(3) Structure computation: linear estimation when 3D points needed.

(4) Bundle Adjustment: final polishing of geometric reprojection error.

3. Reduced measurement matrix

Algebraic cost (computational advantages):

$$d(x_i, E_{ij} x_j) = x_i^T E_{ij} x_j = m_{ij}^T e_{ij}$$

where $\mathbf{e}_{ij}$ → vectorization of matrix $\mathbf{E}_{ij}$. Error $\mathbf{C}_{ege}$ becomes:

$$C_e = \sum_{i \neq j} ||M_{ij} e_{ij}||^2 = \sum_{i \neq j} e_{ij}^T M_{ij}^T M_{ij} e_{ij}$$

Cholesky decomposition of matrix $\Omega = \mathbf{M}^T\mathbf{M}$ provides upper triangular matrix $\mathbf{L}_{9x9}$ equivalent to $\mathbf{M}_{nx9}$ for evaluation of cost function $\mathbf{C}_e$:

$$M^T M = \Omega = L^T L$$

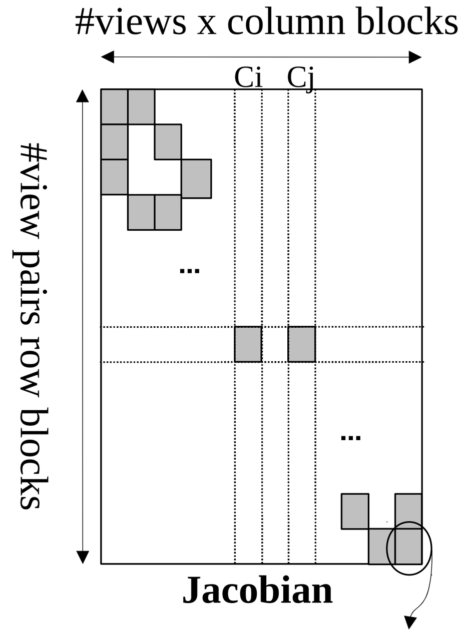
Each $\mathbf{L}_{ij}$ is computed once for each view pair, and reused in subsequent iterative SfM steps substituting matrix $\mathbf{M}_{nx9}$ in the evaluation of $\mathbf{C}_e$.

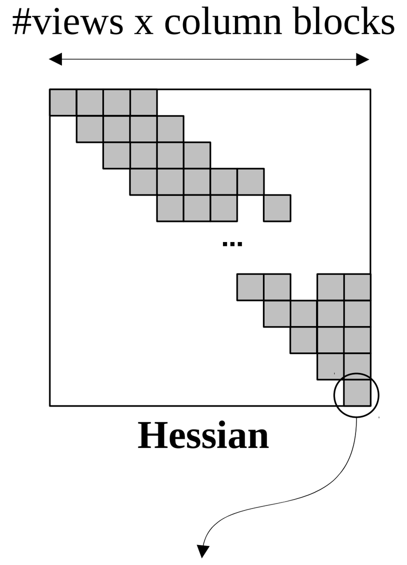
4. Levenberg-Marquardt optimization

→ 3D point coordinates not needed: only 2D view-pair relations. No *Schur complement* required to eliminate 3D point parameters from the LM second level system.

→ Degrees of freedom of the cost function: 6 × *#cams.*

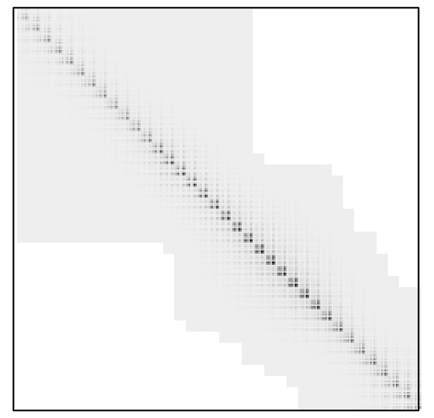
→ GEA Hessian sparsity structure is equivalent to that obtained in BA.





Block size: 6x6.

Block size: nx6, 9x6 after reduction.



Hessian matrix for *dinosaur* dataset.

5. Results

Performance in milliseconds. *Solve* step uses CHOLMOD (sSBA) and MKL (GEA):

Dataset	Set size #Cams	Set size #Pts	sSBA Rest	sSBA Solve	GEA RM	GEA Rest	GEA Solve	GEA LT
trafalgar	256	65k	797	997	282	105	331	957
dubrovnik	356	227k	7024	4351	2977	479	1567	3442
venice	245	199k	6596	604	2837	285	738	3060
venice	427	310k	12387	14480	4708	736	2445	4871

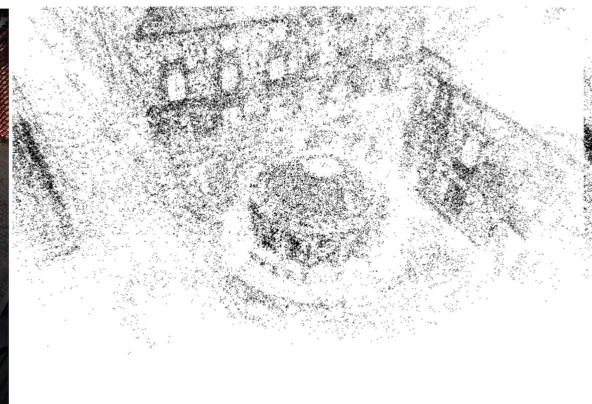
RM → Reduced measurement matrix estimation.

Rest + Solve → Iteration step.

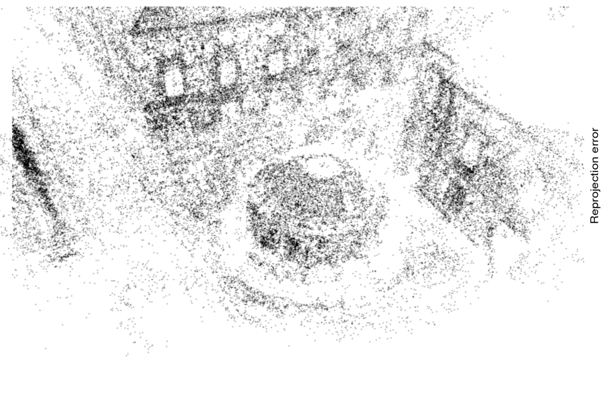
LT → Linear estimation of 3D structure (points).

Optimization of unstructured data-sets:

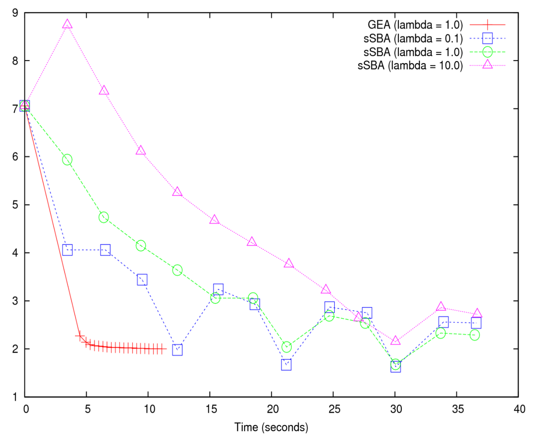
Real



sSBA

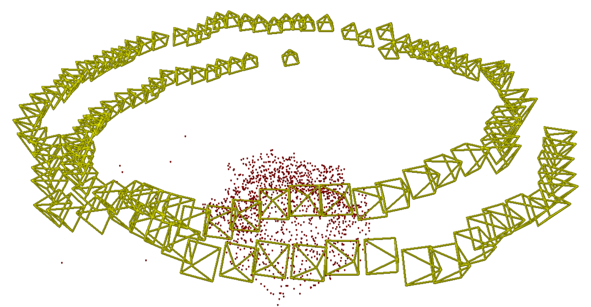


GEA

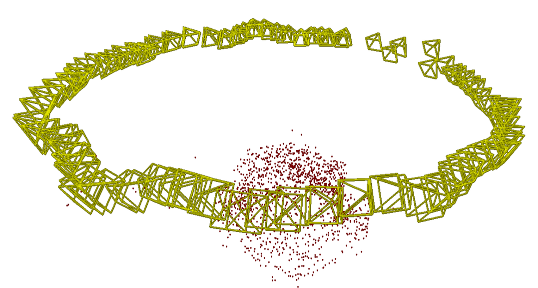


Error reduction vs performance time

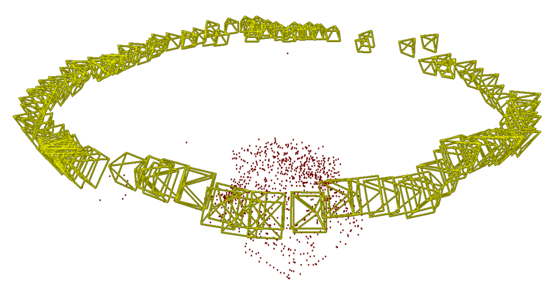
Drift error correction using loop closing information:



Original



sSBA



GEA

Also, convergence basin of GEA is significantly larger, able to recover the correct solution from surprisingly large initial reprojection errors (see Fig. 2 in paper).

URL: http://perception.inf.um.es/gea

1. This work is supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, grants CSD2006-00046 and TIN2009-14475-C04.

2. Departamento de Ingeniería y Tecnología de Computadores, Universidad de Murcia.

3. Departamento de Informática y Sistemas, Universidad de Murcia.