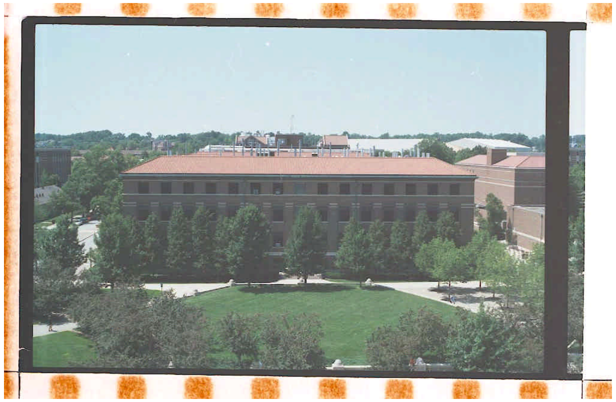
Terrestrial/Close-range Block – Handheld, Ricoh 35mm, f~50mm



chme01

chme02

chme03

chme04

chme05

chme06

Issues: sufficient CP/PP to connect each photo to block, “gimbal lock”, ray intersection angle

Terrestrial Block

chme07

chme08

chme09

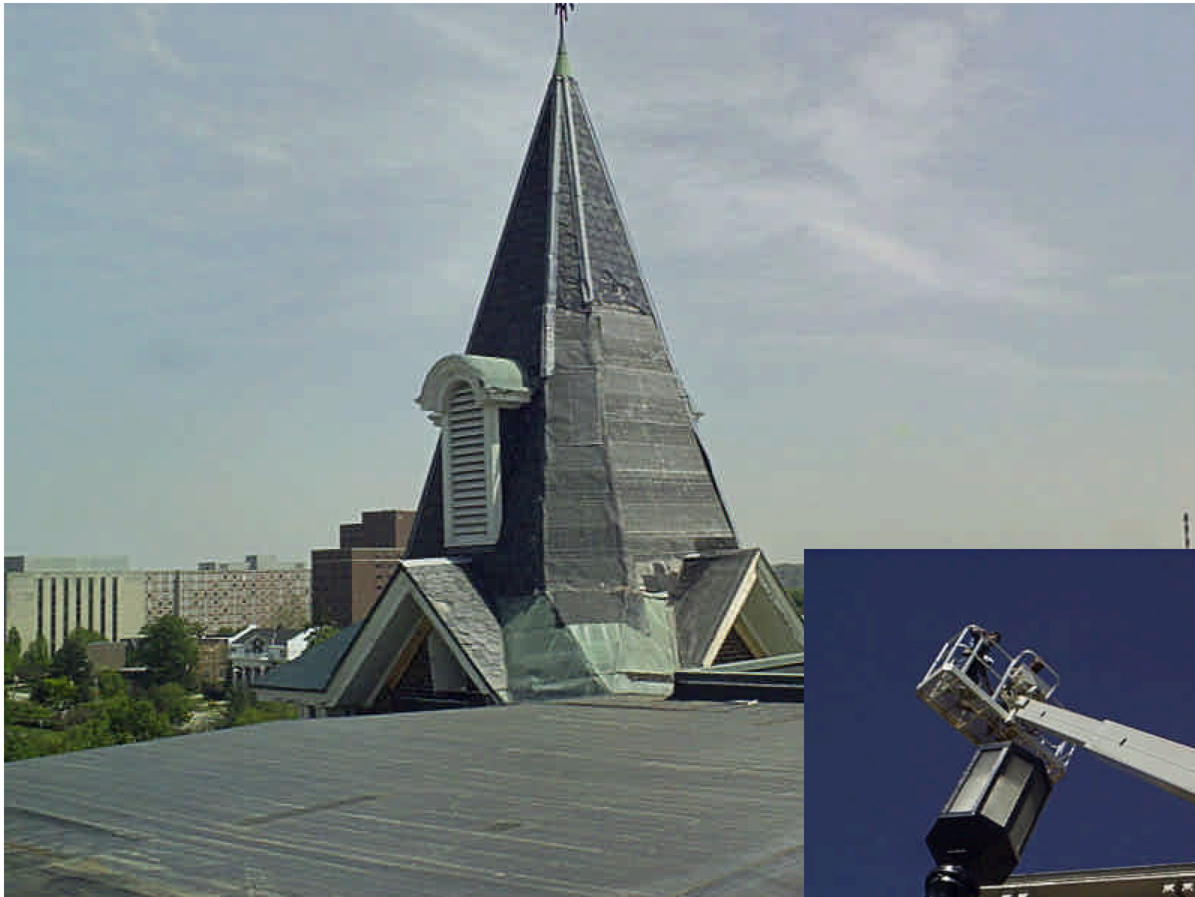
chme10

chme11

chme12

Applications: architecture, restoration, 3D model building/visualization, geopositioning of points not easily seen on "vertical" imagery

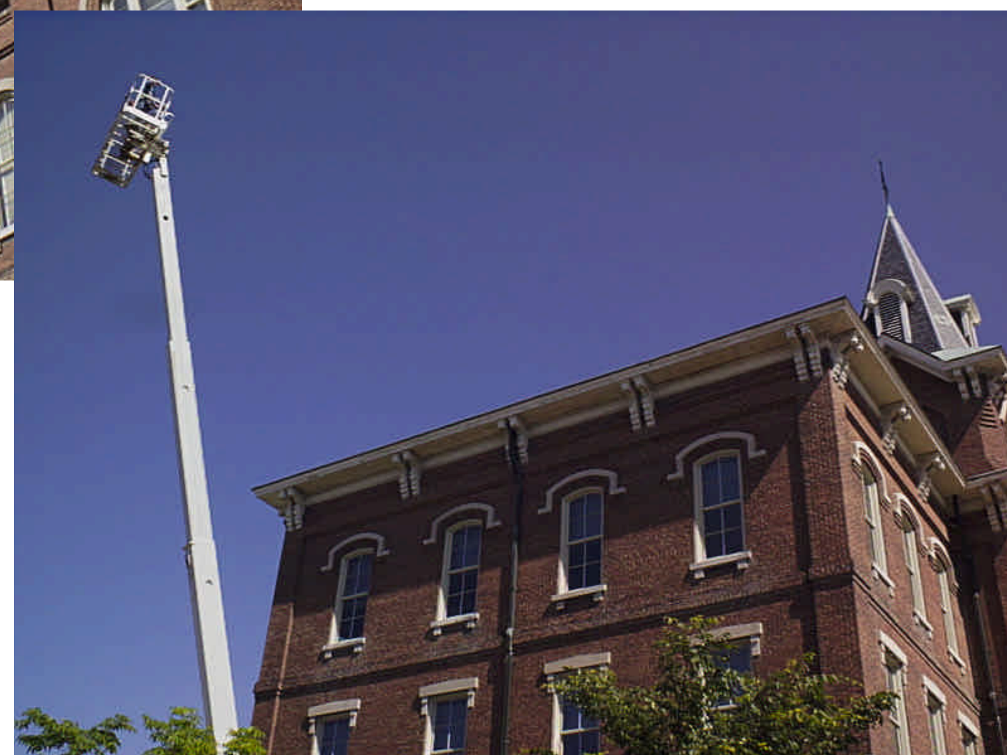
Application: Restoration of University Hall Tower after Wind Damage in 1999



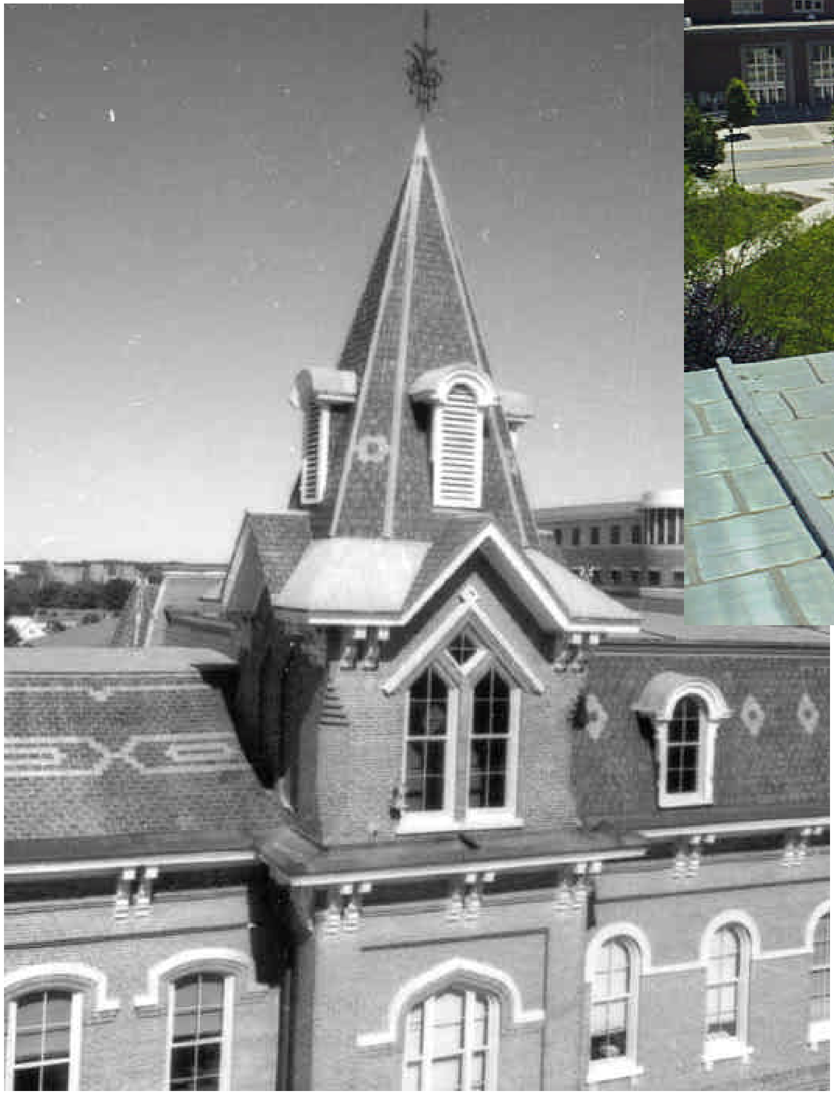
Surface damage, structure weakened, and the whole tower moved

Use close-range photogrammetry to record “as-built” shape of tower for reconstruction”. Use hydraulic lift for camera access, for 360 deg. coverage

Camera used was non-metric Hasselblad (70mm film) with self-calibration in the bundle adjustment

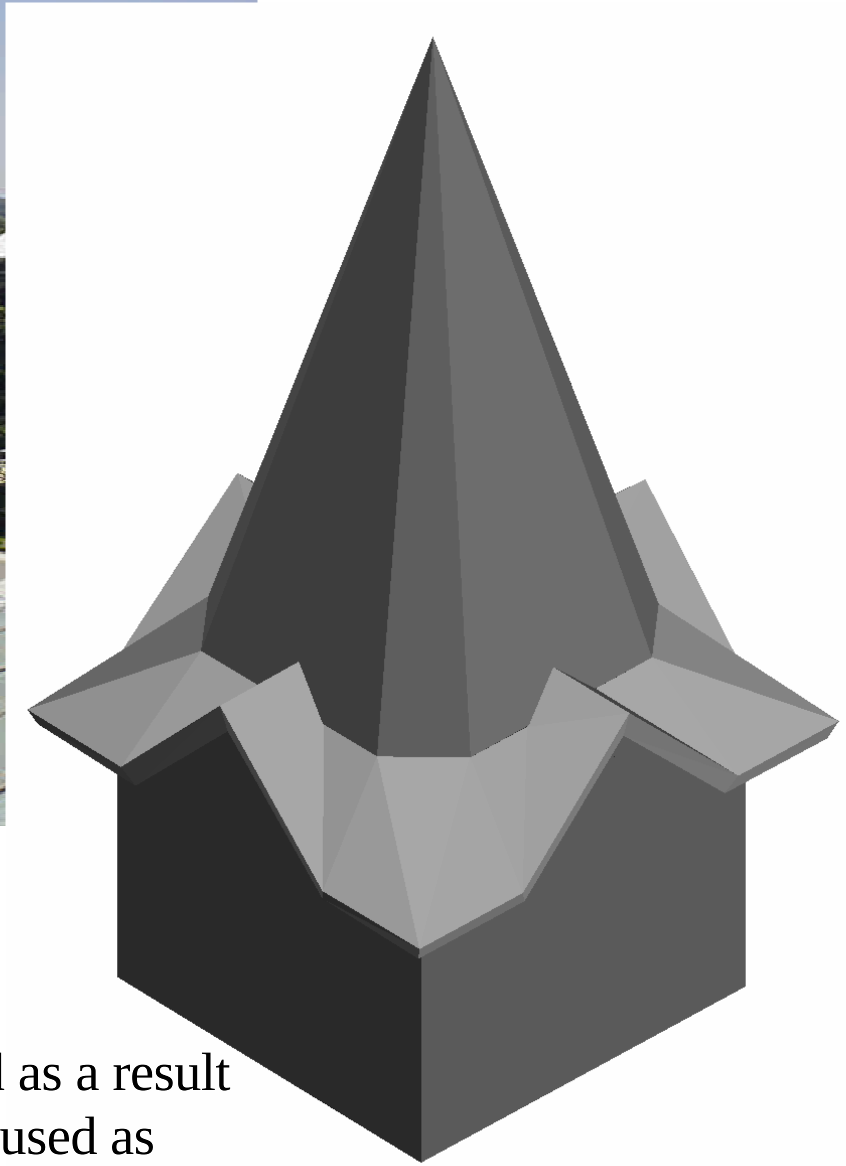


One of the ~15 B&W photos used in the bundle adj.

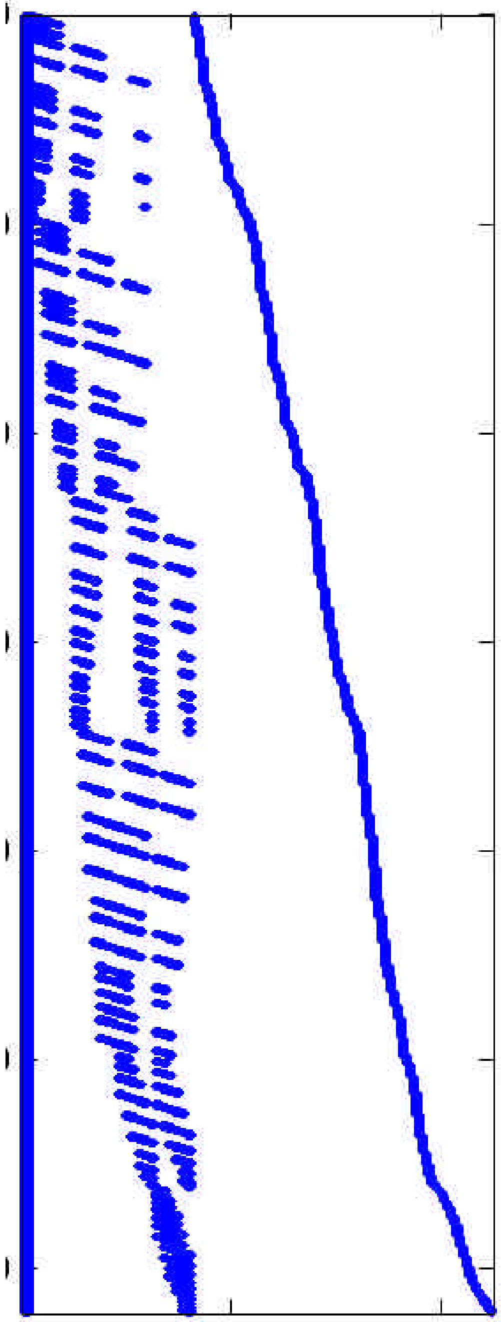


Tying photogrammetric survey to the reference network. Painted target locations determined by leveling and theodolite triangulation from the ground – also visible in the photogrammetric block.

Taping to a control point



3D CAD model produced as a result of the photogrammetry – used as guide for dimensions of new tower



Simulated Block

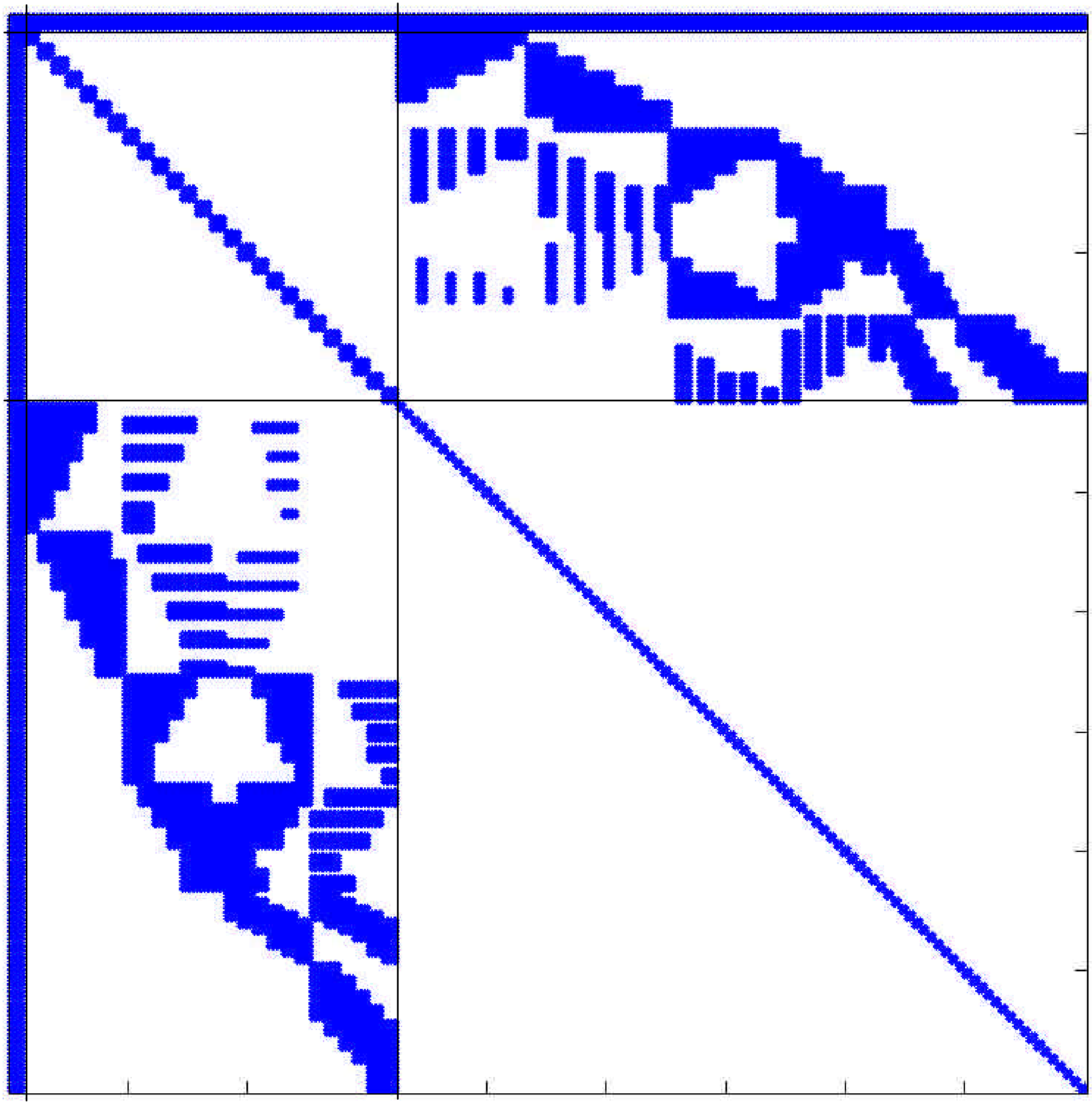
26 photos, 93 pass points, 3 control points

1242 equations, 450 unknowns

Photo & point layout very similar to the Purdue block

Pba.m on sun ultra-10: ~1/2 minute per iteration (all numerical partials, and full matrix inverse)

B Matrix, from command: spy(B)



Simulated Block – normal equations, 450x450, 3 partitions shown: camera internal parameters (6), photo exterior orientation parameters (6 per photo), and ground points (3 per point)

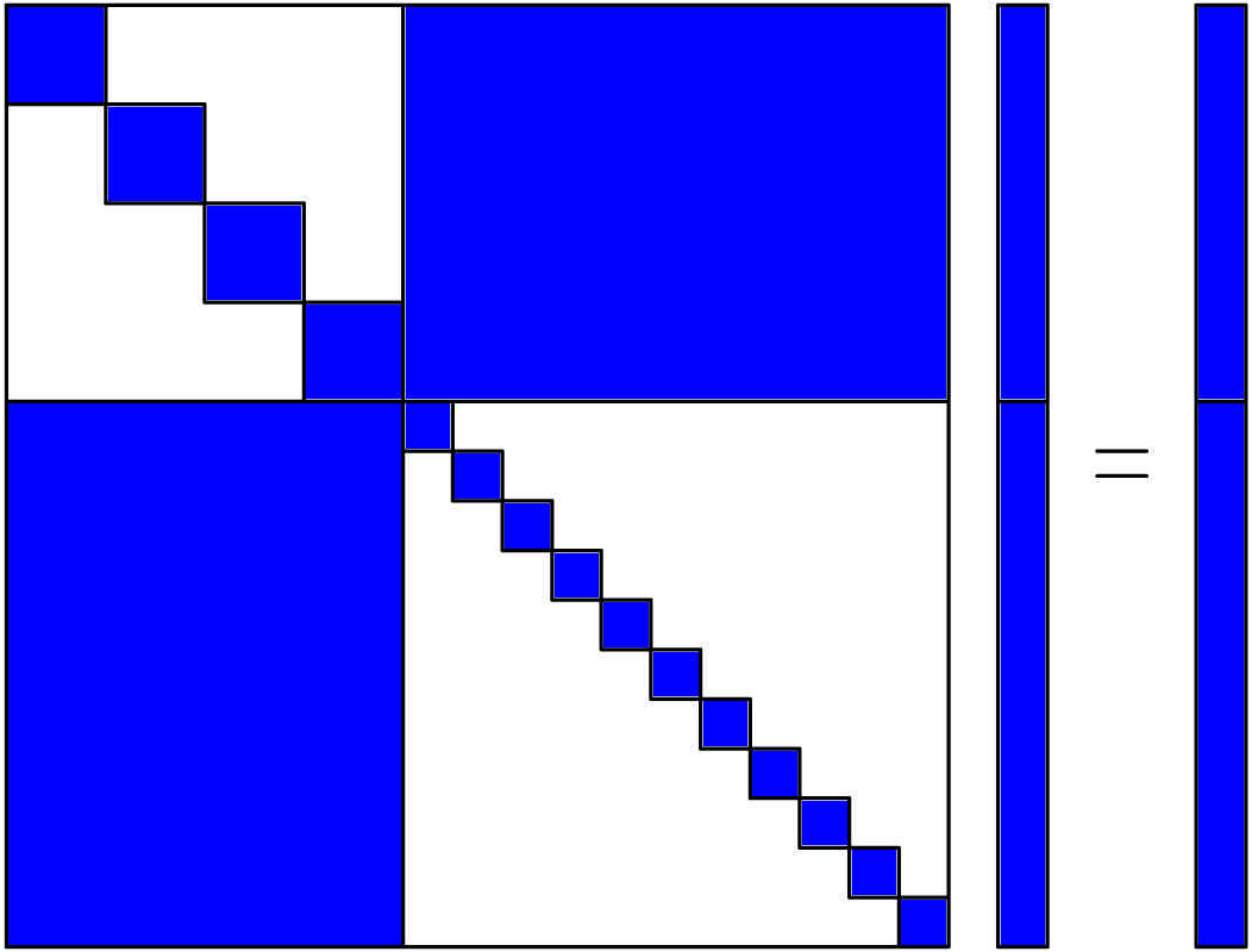
The off diagonal block is nonzero if that point occurs on that photo

The matrix is about 85% zeros.

Let’s look at an efficient block gauss elimination method to derive a set of *reduced normal equations* from the full normal equations

Figure from: spy(N)

Technique to move from full normals to reduced normals: use block gauss elimination to eliminate one point at a time

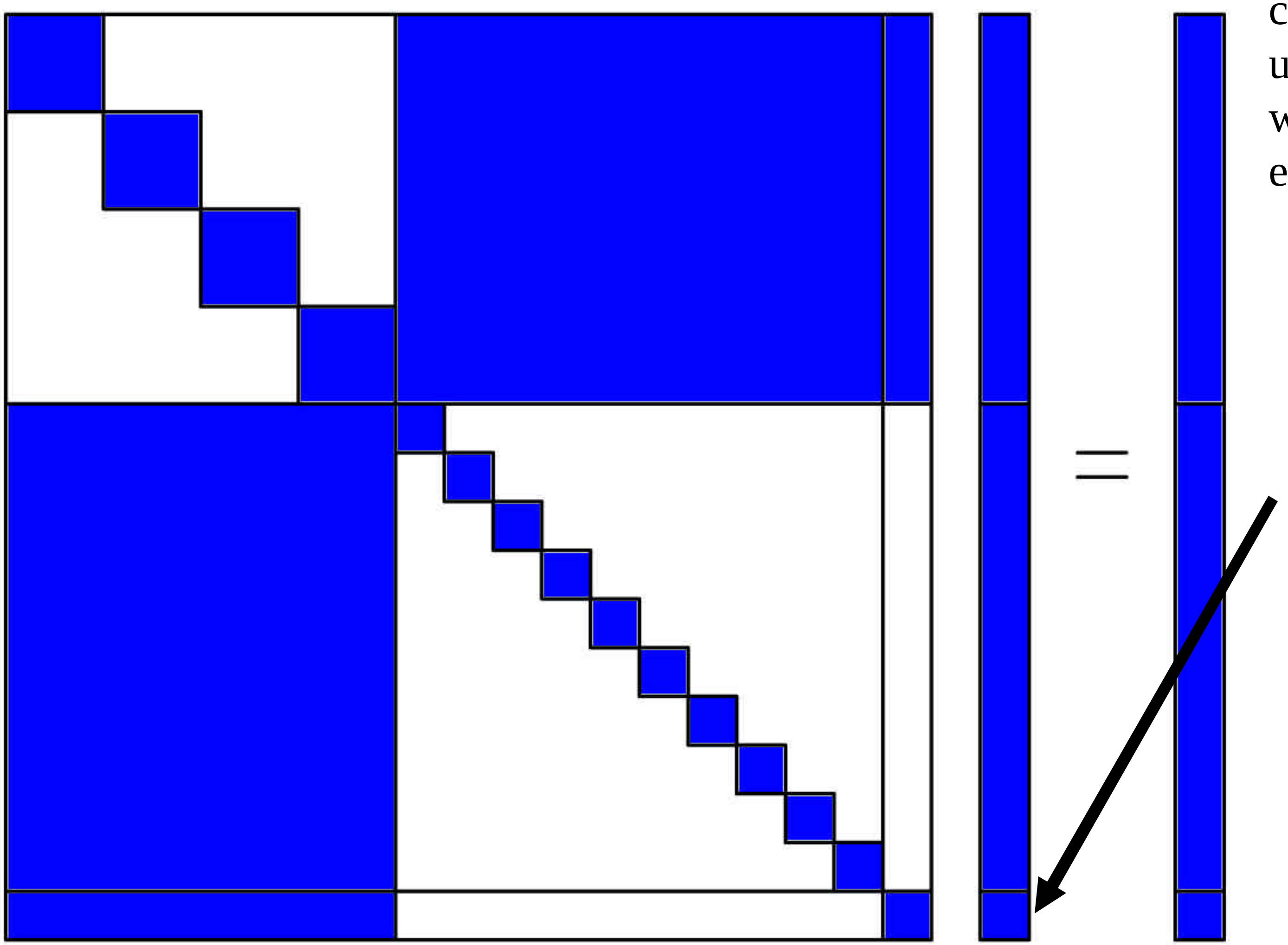


This figure is a schematic representation of

$$\mathbf{N\Delta = t}$$

The off-diagonal partition is often sparse but we will assume full

Block Gauss Elimination

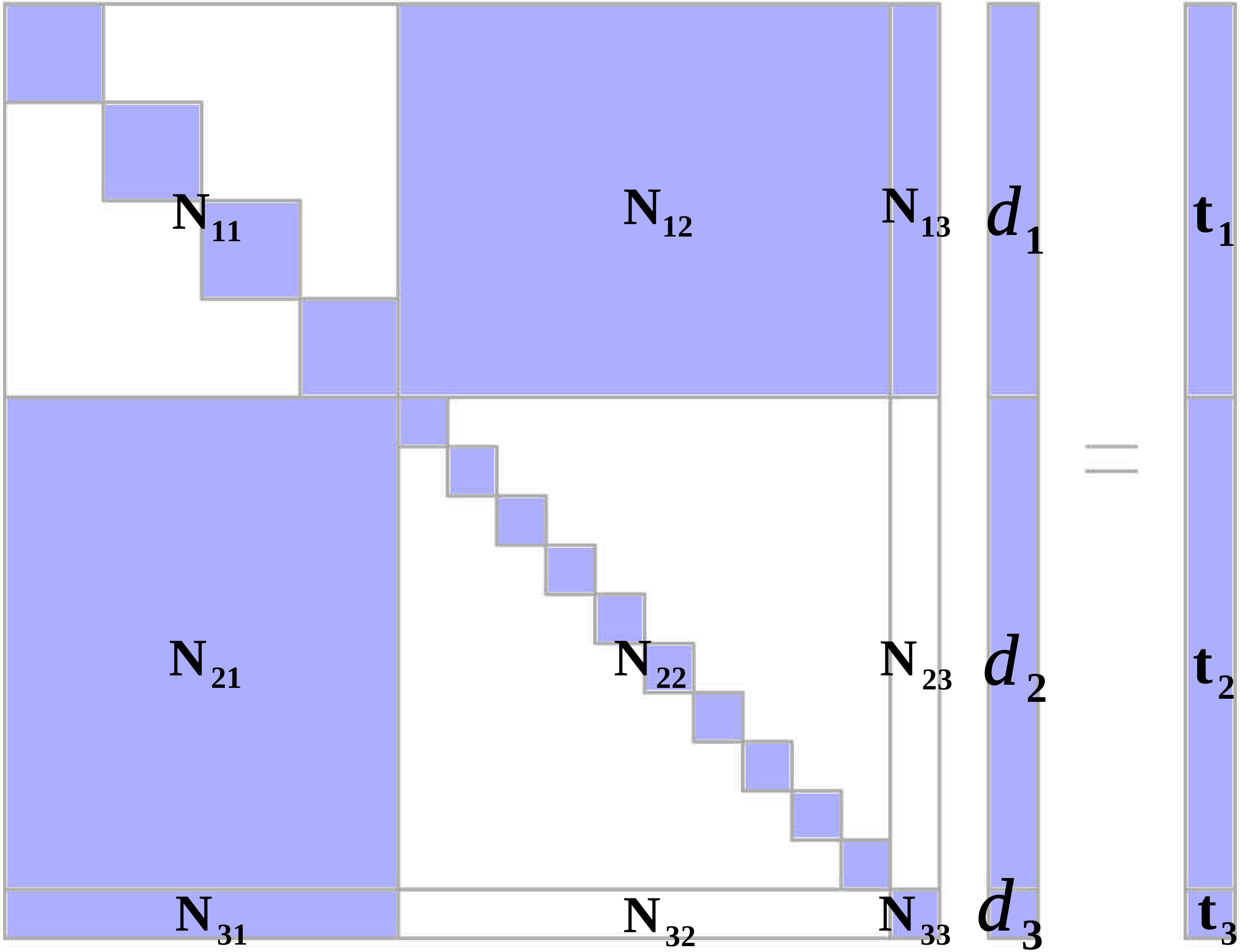


Make a new partition corresponding the unknown(s) that we wish to eliminate

This is the parameter vector partition that we want to eliminate

Recall: forward elimination followed by back substitution for complete solution

Label the partitions. Note that $\mathbf{N}_{23}$ and $\mathbf{N}_{32}$ are zero.



Elimination Step

$$\mathbf{N}_{11}d_1 + \mathbf{N}_{12}d_2 + \mathbf{N}_{13}d_3 = \mathbf{t}_1$$

$$\mathbf{N}_{21}d_1 + \mathbf{N}_{22}d_2 + \mathbf{N}_{23}d_3 = \mathbf{t}_2$$

$$\mathbf{N}_{31}d_1 + \mathbf{N}_{32}d_2 + \mathbf{N}_{33}d_3 = \mathbf{t}_3$$

Eliminate d_3 from eqn. 3, remember

that $\mathbf{N}_{32} = \mathbf{0}$

$$d_3 = \mathbf{N}_{33}^{-1}\left(\mathbf{t}_3 - \mathbf{N}_{31}d_1\right)$$

Remember to save the **N**'s and the **t** from this step so that after we have solved for delta-1 we can come back and solve for delta-3

Now substitute that expression into the first two equations

$$\left(\mathbf{N}_{11} - \mathbf{N}_{13}\mathbf{N}_{33}^{-1}\mathbf{N}_{31}\right)d_1 + \mathbf{N}_{12}d_2 = \mathbf{t}_1 - \mathbf{N}_{13}\mathbf{N}_{33}^{-1}\mathbf{t}_3$$

$$\mathbf{N}_{21}d_1 + \mathbf{N}_{22}d_2 = \mathbf{t}_2$$

Note only changes are in the $\mathbf{N}_{11}$ partition and in the $\mathbf{t}_1$ partition, that is why it is so efficient

Notice that the elimination of a point does not involve the other points – hence they can be accumulated and eliminated right away, never form full $\mathbf{N}_{22}$

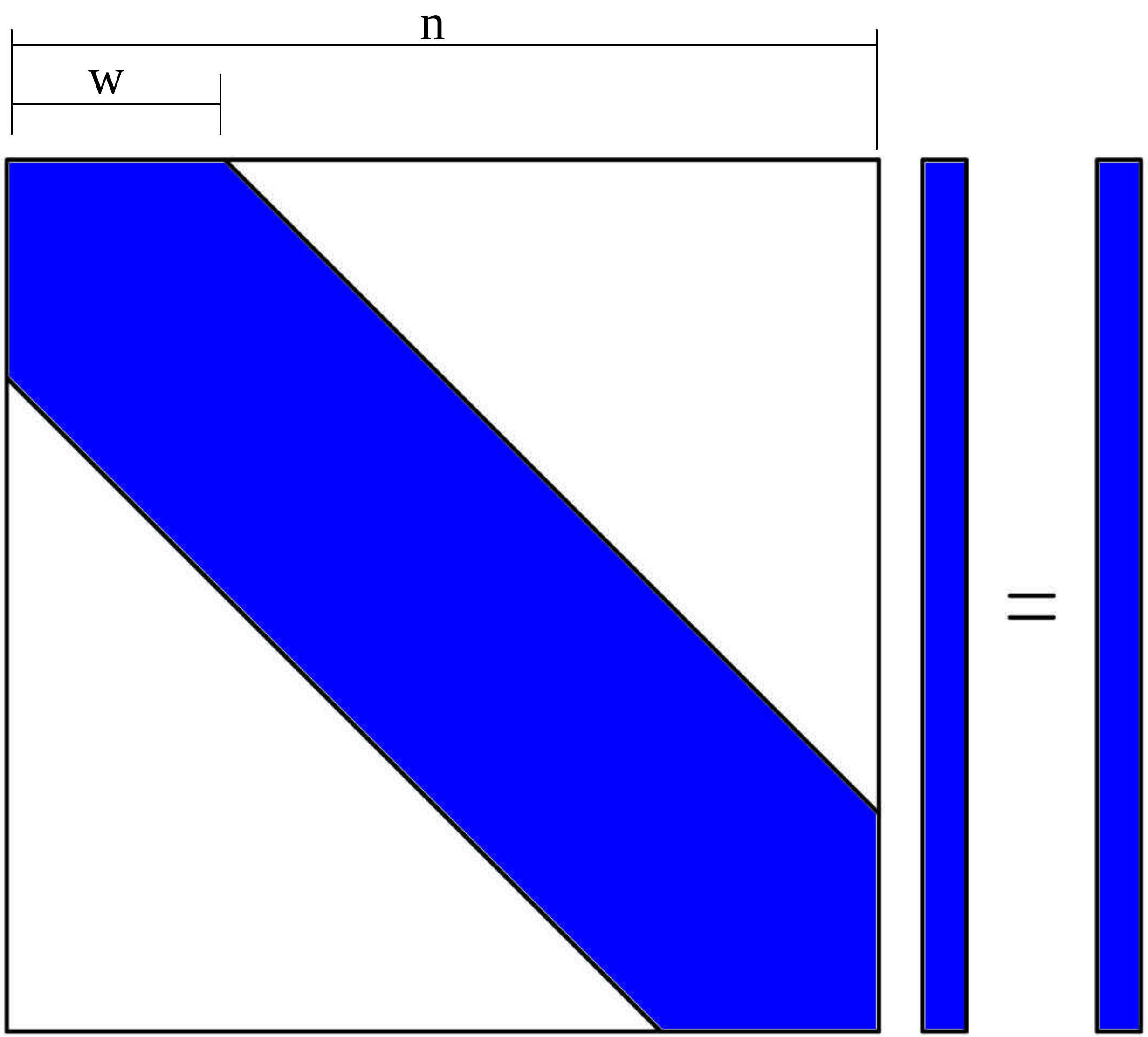
After Eliminating All of the Points

When all points have been eliminated, one by one, by the procedure just described, we are left with the *reduced normal equations* in which now the only remaining unknowns are the photo and camera parameters. The rules, now, for when a block is nonzero: (a) the diagonal blocks are nonzero as before, (b) any off-diagonal block (corresponding to two photos) is nonzero if those two photos share a common point.

With large blocks and parallel flight lines, even this reduced normal equation matrix is still *sparse*. Its structure is banded (or banded-bordered). A similar partitioning plan can be used to successively further reduce this until it is full (number of steps is related to the *bandwidth*).

A famous photogrammetrist, Duane Brown, did much work on the efficient solution of large photogrammetric blocks, and referred to this procedure as *recursive partitioning.*

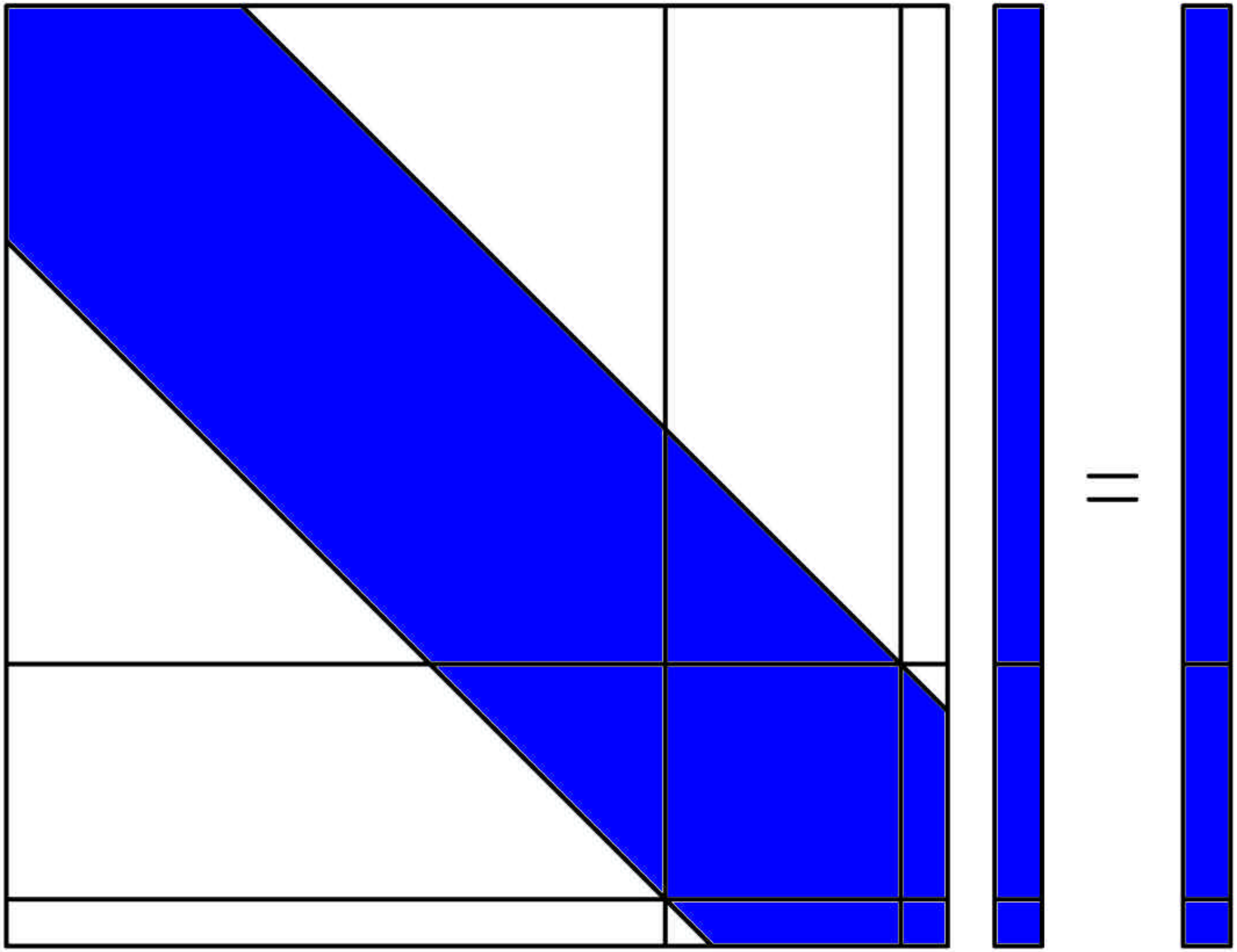
Band Matrices



Solution of System with full matrix takes on the order of n^3 operations. Solution of band system takes on the order of w^2n – that can be *much* less for large n and small w.

Do it by special partitioning to create a zero partition.

Make the partition so that $\mathbf{N}_{13}$ is zero. Then we can efficiently eliminate delta$_3$



Similar to previous elimination step

$$\mathbf{N}_{11}d_1 + \mathbf{N}_{12}d_2 + \mathbf{N}_{13}d_3 = \mathbf{t}_1$$
$$\mathbf{N}_{21}d_1 + \mathbf{N}_{22}d_2 + \mathbf{N}_{23}d_3 = \mathbf{t}_2$$
$$\mathbf{N}_{31}d_1 + \mathbf{N}_{32}d_2 + \mathbf{N}_{33}d_3 = \mathbf{t}_3$$

Eliminate d_3 from eqn. 3, remember

that $\mathbf{N}_{31} = \mathbf{0}$

$$d_3 = \mathbf{N}_{33}^{-1}\left(\mathbf{t}_3 - \mathbf{N}_{32}d_2\right)$$

Now substitute that expression into the

first two equations

$$\mathbf{N}_{11}d_1 + \mathbf{N}_{12}d_2 = \mathbf{t}_1$$
$$\mathbf{N}_{21}d_1 + \left(\mathbf{N}_{22} - \mathbf{N}_{23}\mathbf{N}_{33}^{-1}\mathbf{N}_{32}\right)d_2 = \mathbf{t}_2 - \mathbf{N}_{23}\mathbf{N}_{33}^{-1}\mathbf{t}_3$$

Notice all activity takes place within the band

This is the forward elimination step. Do it many times, saving intermediate results. Then do back subsititution, recalling saved results until the full banded system is solved. That gets you the photo parameters, then do back substitution for the eliminated points, and you are done – for *this iteration !*

Solution Strategy for Block Adjustment

- Process data by point
- When all contributions (equations) for a point have been constructed, eliminate it, and save intermediate results
- After all points have been processed and eliminated, you have left only the camera and photo parameters, which are banded or band-bordered
- Use band matrix processing to efficiently solve for camera/photo parameters
- Retrieve saved intermediate results and solve for all of the ground points
- Finished this iteration – keep going until you converge!