Structure from Perspective Images with Occlusions and Outliers

Daniel Martinec and Tomáš Pajdla

Center for Machine Perception
Department of Cybernetics, Faculty Electrical Engineering
Czech Technical University, Prague

Outline

INTRODUCTION

- Problem formulation, previous work, our contribution

PART I: No Outliers

- Used techniques
  - projective depth estimation
  - filling the missing elements
  - combination of the above
- Experiments

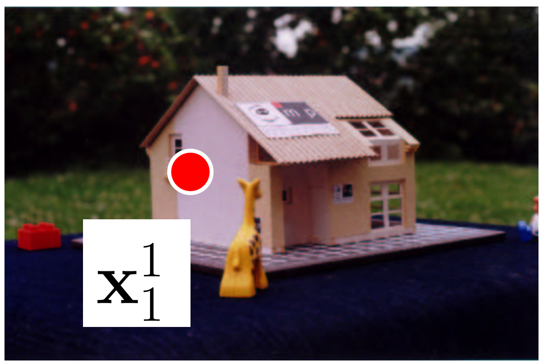
PART II: Outlier Detection

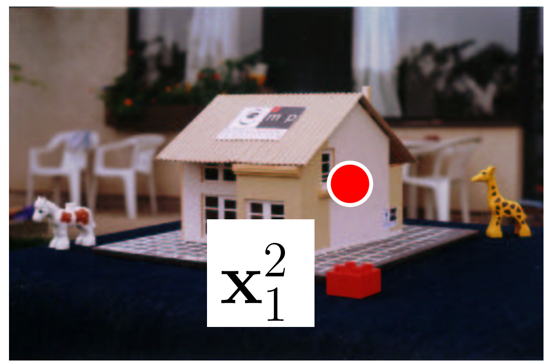
- New idea & algorithm
- Experiments

Goal

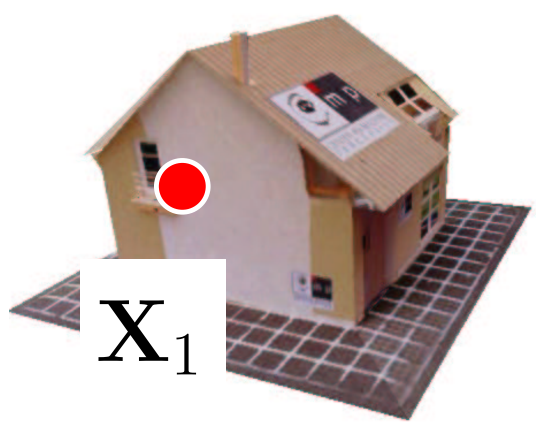
projective 3D-reconstruction from perspective images

with occlusions and outliers





$$\Downarrow$$



Contribution

Previous & Related work

[1] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. In *IJCV(9)2*, 1992.

[2] D. Jacobs. Linear fitting with missing data: Applications to structure from motion and to characterizing intensity images. In *CVPR*, 1997.

[3] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *ECCV*, 1996.

[4] D. Q. Huynh and A. Heyden. Outlier detection in video sequences under affine projection. In *CVPR*, 2001.

Main Contribution

- Generalization of Jacobs [2]
- combination with Sturm & Triggs [3]
- extension for outliers

to obtain 3D-reconstruction from perspective images with occlusions and outliers by factorization

PART I: No Outliers

Problem Formulation & Solution

Perspective camera projection: $\lambda_p^i \underbrace{\mathbf{x}_p^i}_{3\times 1} = \underbrace{\mathtt{P}^i}_{3\times 4} \underbrace{\mathbf{X}_p}_{4\times 1}$

$$\underbrace{\begin{bmatrix} \lambda_1^1\mathbf{x}_1^1 & \lambda_2^1\mathbf{x}_2^1 & \dots & \lambda_n^1\mathbf{x}_n^1 \\ \times & \lambda_2^2\mathbf{x}_2^2 & & \times \\ \vdots & & \ddots & \vdots \\ \lambda_1^m\mathbf{x}_1^m & \times & \dots & \lambda_n^m\mathbf{x}_n^m \end{bmatrix}}_{\mathtt{R}} = \underbrace{\begin{bmatrix} \mathtt{P}^1 \\ \mathtt{P}^2 \\ \vdots \\ \mathtt{P}^m \end{bmatrix}}_{3m \times 4} \underbrace{[\mathbf{X}_1 \mathbf{X}_2 \dots \mathbf{X}_n]}_{4 \times n}$$

rescaled measurement matrix — motion — structure

New Algorithm

1. *Estimate (some)* λ_j^i using e.g. the epipolar geometry [3].
2. *Fill* $\times$ using the extension of [2] for perspective cameras.

   Repeat 1 and 2 until $\mathtt{R}$ is complete.
3. *Factorize* [1] complete $\mathtt{R}$.

$\longrightarrow$ Steps 1, 2: main components

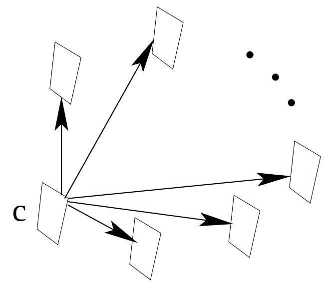
Projective Depth Estimation (Sturm & Triggs [3])

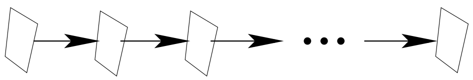
uses fundamental matrices

$$\lambda_p^i = \frac{(\mathbf{e}^{ij} \wedge \mathbf{x}_p^i) \cdot (\mathtt{F}^{ij}\ \mathbf{x}_p^j)}{\|\ \mathbf{e}^{ij} \wedge \mathbf{x}_p^i\ \|^2}\ \lambda_p^j$$

$\wedge$ . . . cross-product

- **Alternatives:** 1. central image 2. sequence





New Filling for Perspective Cameras

- extension of filling for affine cameras ($\lambda_p^i = 1$) by Jacobs [2]
- fill $\mathtt{R}$ so that rank $\mathtt{R} = 4$ (rank $\mathtt{R} = 4$ because $\mathtt{R} = \underbrace{\mathtt{P}}_{3m\times 4}\underbrace{\mathtt{X}}_{4\times n}$)

a 4-tuple of "LI" columns

$$\mathtt{B}_k = \begin{bmatrix} ?\ \mathbf{x}_1^1 & \lambda_2^1\mathbf{x}_2^1 & \lambda_3^1\mathbf{x}_3^1 & \lambda_4^1\mathbf{x}_4^1 \\ \lambda_1^2\mathbf{x}_1^2 & \lambda_2^2\mathbf{x}_2^2 & \lambda_3^2\mathbf{x}_3^2 & \lambda_4^2\mathbf{x}_4^2 \\ \vdots & & & \\ \lambda_1^m\mathbf{x}_1^m & \lambda_2^m\mathbf{x}_2^m & \lambda_3^m\mathbf{x}_3^m & \times \end{bmatrix} = \begin{bmatrix} ?\ x_1^1 & \lambda_2^1 x_2^1 & \lambda_3^1 x_3^1 & \lambda_4^1 x_4^1 \\ ?\ y_1^1 & \lambda_2^1 y_2^1 & \lambda_3^1 y_3^1 & \lambda_4^1 y_4^1 \\ ?\ w_1^1 & \lambda_2^1 w_2^1 & \lambda_3^1 w_3^1 & \lambda_4^1 w_4^1 \\ \lambda_1^2 x_1^2 & \lambda_2^2 x_2^2 & \lambda_3^2 x_3^2 & \lambda_4^2 x_4^2 \\ \lambda_1^2 y_1^2 & \lambda_2^2 y_2^2 & \lambda_3^2 y_3^2 & \lambda_4^2 y_4^2 \\ \lambda_1^2 w_1^2 & \lambda_2^2 w_2^2 & \lambda_3^2 w_3^2 & \lambda_4^2 w_4^2 \\ \vdots & & & \\ \lambda_1^m x_1^m & \lambda_2^m x_2^m & \lambda_3^m x_3^m & \times \\ \lambda_1^m y_1^m & \lambda_2^m y_2^m & \lambda_3^m y_3^m & \times \\ \lambda_1^m w_1^m & \lambda_2^m w_2^m & \lambda_3^m w_3^m & \times \end{bmatrix}$$

linear hull of all possible fillings

$$\mathcal{B}_k = Span\left(\begin{bmatrix} 0 & x_1^1 & \lambda_2^1 x_2^1 & \lambda_3^1 x_3^1 & \lambda_4^1 x_4^1 & 0 & 0 & 0 \\ 0 & y_1^1 & \lambda_2^1 y_2^1 & \lambda_3^1 y_3^1 & \lambda_4^1 y_4^1 & 0 & 0 & 0 \\ 0 & w_1^1 & \lambda_2^1 w_2^1 & \lambda_3^1 w_3^1 & \lambda_4^1 w_4^1 & 0 & 0 & 0 \\ \lambda_1^2 x_1^2 & 0 & \lambda_2^2 x_2^2 & \lambda_3^2 x_3^2 & \lambda_4^2 x_4^2 & 0 & 0 & 0 \\ \lambda_1^2 y_1^2 & 0 & \lambda_2^2 y_2^2 & \lambda_3^2 y_3^2 & \lambda_4^2 y_4^2 & 0 & 0 & 0 \\ \lambda_1^2 w_1^2 & 0 & \lambda_2^2 w_2^2 & \lambda_3^2 w_3^2 & \lambda_4^2 w_4^2 & 0 & 0 & 0 \\ \vdots & & & & & & & \\ \lambda_1^m x_1^m & 0 & \lambda_2^m x_2^m & \lambda_3^m x_3^m & 0 & \mathbf{1} & 0 & 0 \\ \lambda_1^m y_1^m & 0 & \lambda_2^m y_2^m & \lambda_3^m y_3^m & 0 & 0 & \mathbf{1} & 0 \\ \lambda_1^m w_1^m & 0 & \lambda_2^m w_2^m & \lambda_3^m w_3^m & 0 & 0 & 0 & \mathbf{1} \end{bmatrix}\right)$$

- $\mathcal{B}_k$ contains filled $\mathtt{B}_k$ (also filled $\mathtt{R}$) $\longrightarrow$ constraint on $\mathtt{R}$
- intersection of many $\mathcal{B}_k$ $\longrightarrow$ fill $\mathtt{R}$

Heuristics Guiding the Filling of $\mathtt{R}$

- many alternatives for computing λ_p^i
- noise $\longrightarrow$ alternatives not equivalent
- heuristics $\longrightarrow$ good reconstruction

Proposition 1.
The repr. error decreases with the increase of the number of known λs.

very many data missing $\longrightarrow$ depth estimation and filling must be repeated

Observation 1.
The repr. error decreases with the increase of image points that are filled in one step.

Conclusion
Choose the alternative which fills the most data and using which the most λs are computed.

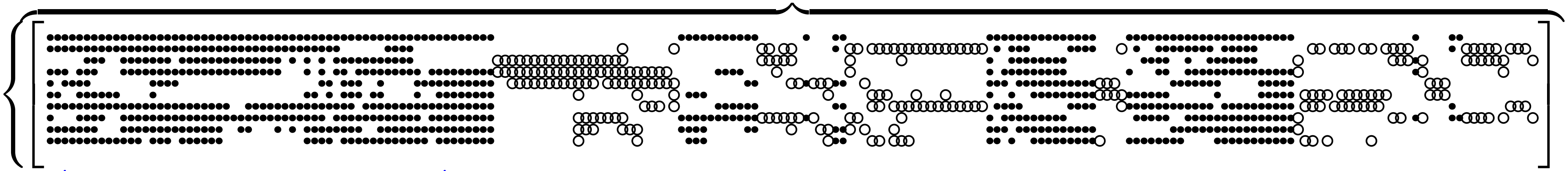
Wide Base-Line Stereo

LM = lin. method, BA = bundle adjustment

	Scene *House*	10 images [2952×2003]	
	Point detection	manual, 203 points in space	
	Depth estimation	central image No. 1	
	Amount of missing data		**47.83 %**
LM	Mean error per image point [pxl]		**3.91**
LM + BA			**1.44**

203 points

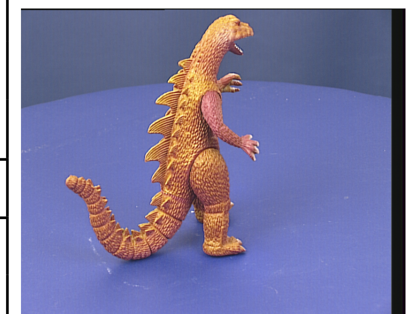
10 images

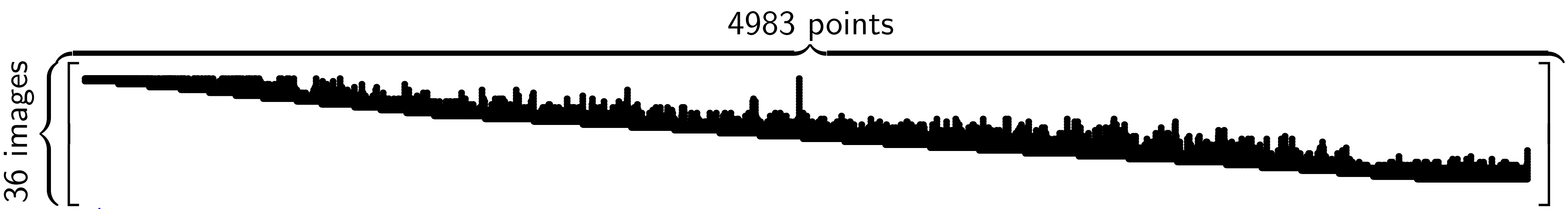


"$\bullet$" $\lambda_p^i \longleftarrow$ F (75.7 %), "○" $\lambda_p^i \longleftarrow \mathcal{B}$ (24.3 %), " " occlusion

Dense Sequence (Oxford)

	Scene *Dinosaur (Oxford)*	36 images [720×576]
	Point detection	Harris' operator, 4983 points in space
	Depth estimation	sequence
	Amount of missing data	**90.84 %**
LM	Mean error per image point [pxl]	**1.76**
LM + BA		**0.64**





"●" $\lambda_p^i \longleftarrow$ F (100.0 %), " " occlusion

PART II: Outlier Detection

Problem Formulation & Related Work

Perspective camera projection: $\lambda_p^i \underbrace{\mathbf{x}_p^i}_{3\times 1} = \underbrace{\mathtt{P}^i}_{3\times 4} \underbrace{\mathbf{X}_p}_{4\times 1}$

$\mathbf{x}_p^i$ . . . outliers

$$\underbrace{\begin{bmatrix} \lambda_1^1\mathbf{x}_1^1 & \lambda_2^1\mathbf{x}_2^1 & \dots & \lambda_n^1\mathbf{x}_n^1 \\ \times & \lambda_2^2\mathbf{x}_2^2 & & \times \\ \vdots & & \ddots & \vdots \\ \lambda_1^m\mathbf{x}_1^m & \times & \dots & \lambda_n^m\mathbf{x}_n^m \end{bmatrix}}_{\mathtt{R}} = \underbrace{\begin{bmatrix} \mathtt{P}^1 \\ \mathtt{P}^2 \\ \vdots \\ \mathtt{P}^m \end{bmatrix}}_{3m \times 4} \underbrace{[\mathbf{X}_1 \mathbf{X}_2 \dots \mathbf{X}_n]}_{4 \times n}$$

rescaled measurement matrix | motion | structure

Previous & Related Work

[1] D. Martinec and T. Pajdla. Structure from many perspective images with occlusions. In *ECCV*, 2002.

[2] D. Q. Huynh and A. Heyden. Outlier detection in video sequences under affine projection. In *CVPR*, 2001.

Main Idea

Assumption

One dominant motion and random independent outliers

Main Idea

Minimal configurations of points in *triples* of images are sufficient to validate inliers reliably.

$\Longrightarrow$ simplified RANSAC: not some large but some minimal support is sufficient

1. Choose a triple of images and six common points in them in random $\longrightarrow \mathcal{T}$
2. If $\mathcal{T}$ consistent with enough (9) correspondences $\longrightarrow$ inliers

Repeat 1 and 2 until there is enough inliers

$\longrightarrow$ estimate reconstruction using our method with occlusions [1] (outliers$\rightarrow$occlusions)

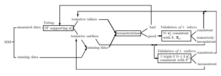
Outlier Detection Algorithm

1. **Initial set of inliers.** Using $\mathcal{T}$s.
2. **Reconstruction** from inliers using [1].
3. **Check** all data whether it is consistent with the reconstruction.

   In each column, p, of $\mathtt{R}$:

   (a) Random triple of image points, $\mathtt{P} \longrightarrow \mathbf{X}_p$
   (b) If repr. error small $\longrightarrow$ inliers

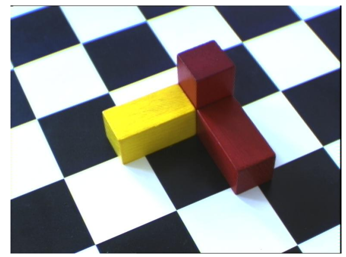
   Repeat (a) and (b) until the column is sufficiently sampled
4. **Iteration** of Steps 2 and 3 while any new inlier appeared

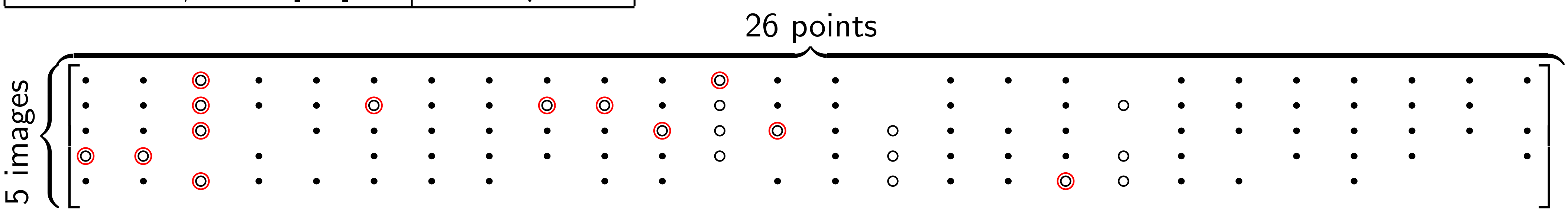


Experiments with Outliers

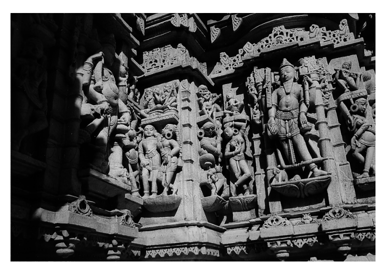
"○" outlier, " " occlusion, "•" $\lambda_p^i \longleftarrow \mathtt{F}$, "○" $\lambda_p^i \longleftarrow \mathcal{B}$

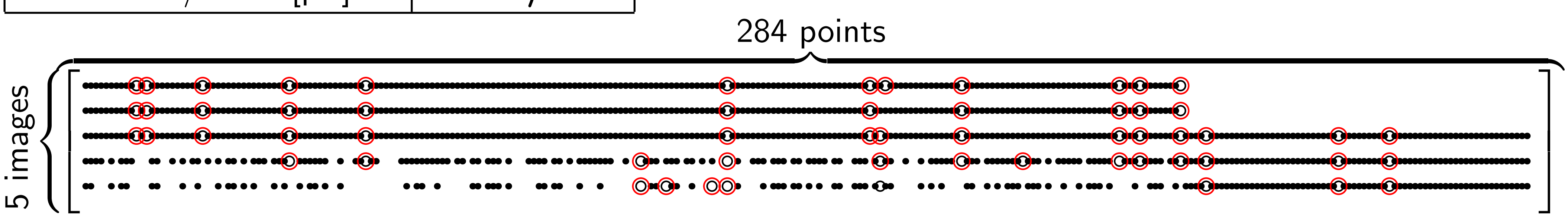
Cubes	5 images [768×576]
Corresp. / outl. det.	manual / 5
Depth estimation	central image No. 1
All / cont. / p. val. tracks	26 / 10 / 9
Mean error / outl. [pxl]	**0.22 / 6.93**





Temple (Leuven)	5 images [867×591]
Corresp. / outl. det.	Harris' operator / 2
Depth estimation	sequence
All / cont. / p. val. tracks	284 / 20 / 6
Mean error / outl. [pxl]	**0.27 / 3.64**





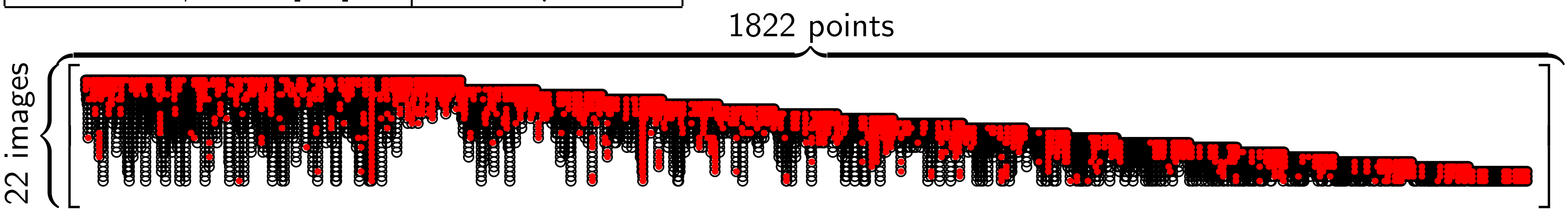
When Discontinuous Tracks Present

track = correspondence = column of $\mathtt{R}$ = projections of the pth world point into all images

registering subtracks & joining the overlapping ones

use disjoined subtracks in reconstruction separately

Castle (Leuven)	22 images [768×576]
Corresp. / outl. det.	Harris' operator / 1
Depth estimation	sequence
All / cont. / p. val. tracks	1822 / 716 / 338
Mean error / outl. [pxl]	**0.22 / 11.97**



Conclusion

- A **new algorithm** for a general situation
  - **perspective** camera
  - **occlusions**
  - **outliers**
- Experiments
  - real scenes: applicable in wide base-line as well as on sequences

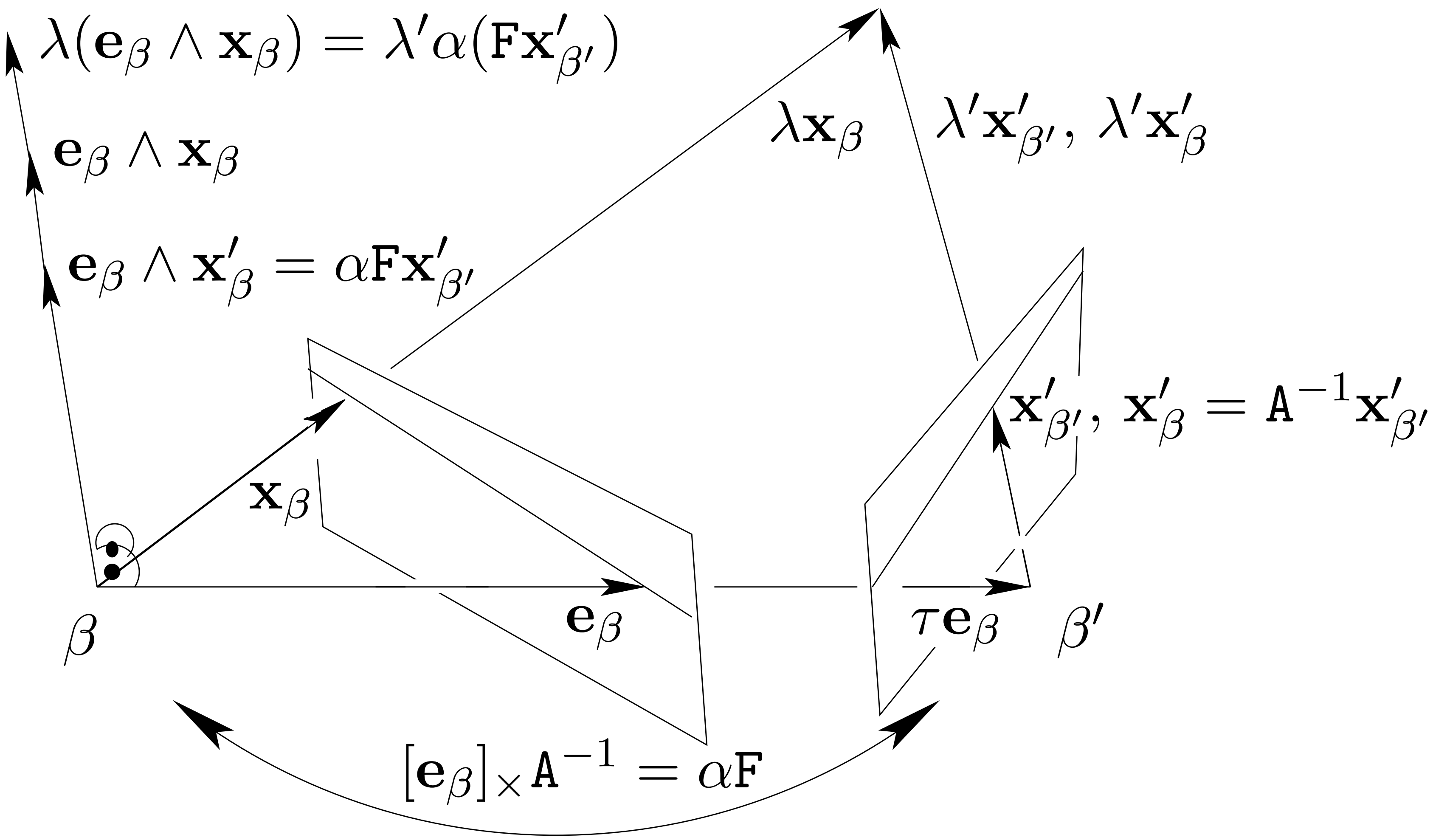
Initial Set of Inliers

Let μ denote the size of the minimal consistent set, $\mu > 6$.

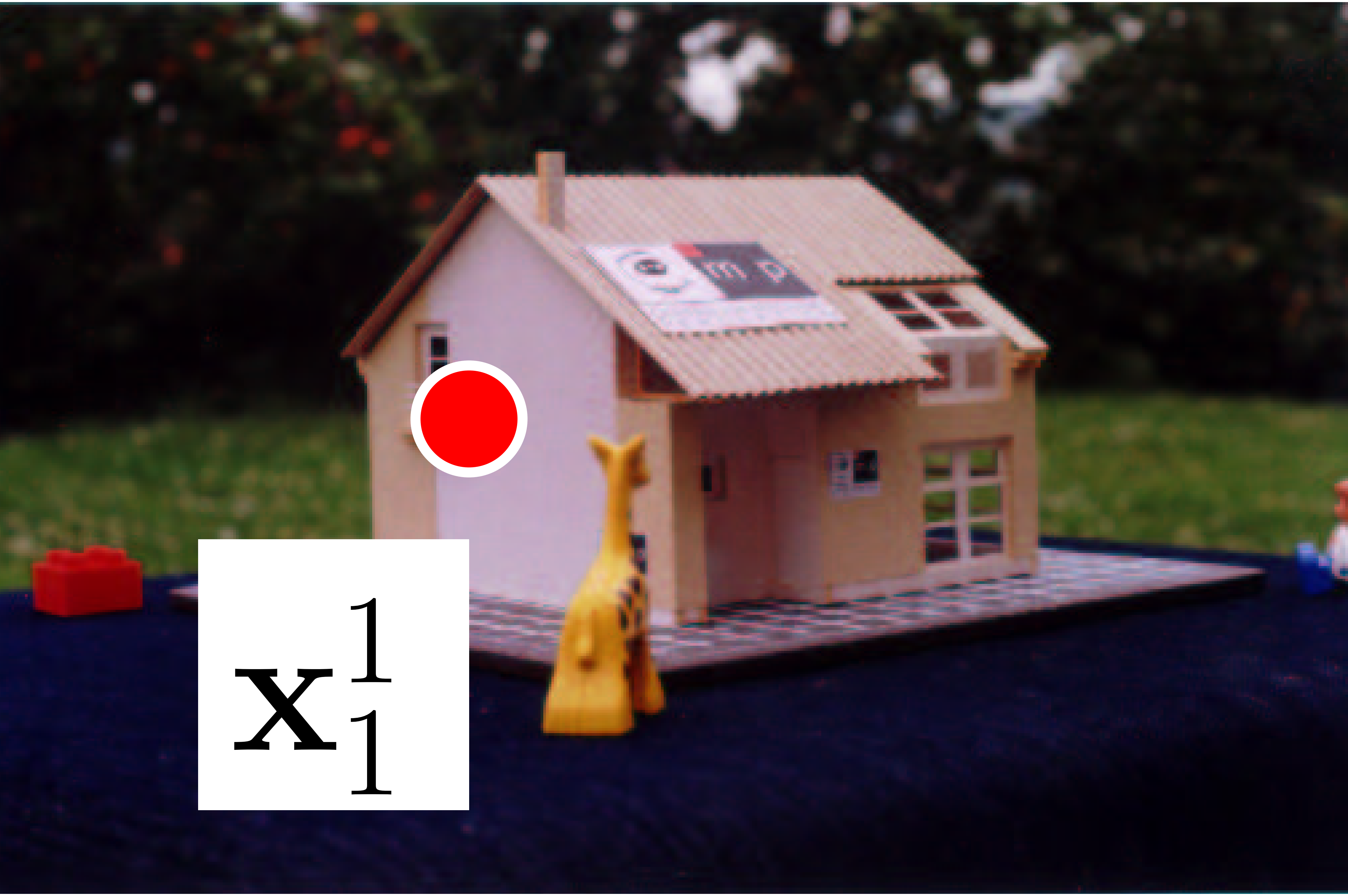
1. Choose randomly a triple of images so that there are at least μ common points in these images. Let $\mathbf{i} = \{i, j, k\}$ denote the index set of the chosen images. Let $\mathbf{p}$ denote the index set of the points visible in images $\mathbf{i}$.
2. In images $\mathbf{i}$, choose randomly 6 common points in a non-degenerate configuration and estimate $\mathcal{T}$ and camera matrices $\mathtt{P}^i$, $i \in \mathbf{i}$. There will be one or three real solutions.
3. **Finding the consistent set with $\mathcal{T}$.**
   (a) Estimate 3D points $\mathbf{X}_p$, $p \in \mathbf{p}$, using $\mathtt{P}^{\mathbf{i}}$.
   (b) Calculate the reprojection errors as
   $e_p = \max_{i \in \mathbf{i}} d(\mathbf{x}_p^i, \mathtt{P}^i \mathbf{X}_p)$
   (c) Compute the number of inliers consistent with $\mathtt{P}^{\mathbf{i}}$ ($\mathcal{T}$) by the number of correspondences for which $e_p < t$.
   (d) If there are three real solutions for $\mathcal{T}$ the number of inliers is computed for each solution, and the solution with most inliers retained.
4. **Voting.** If size of the consistent set is at least μ, then its image points except those used to estimate $\mathcal{T}$ are (i) given a vote and (ii) used for updating the tracks.

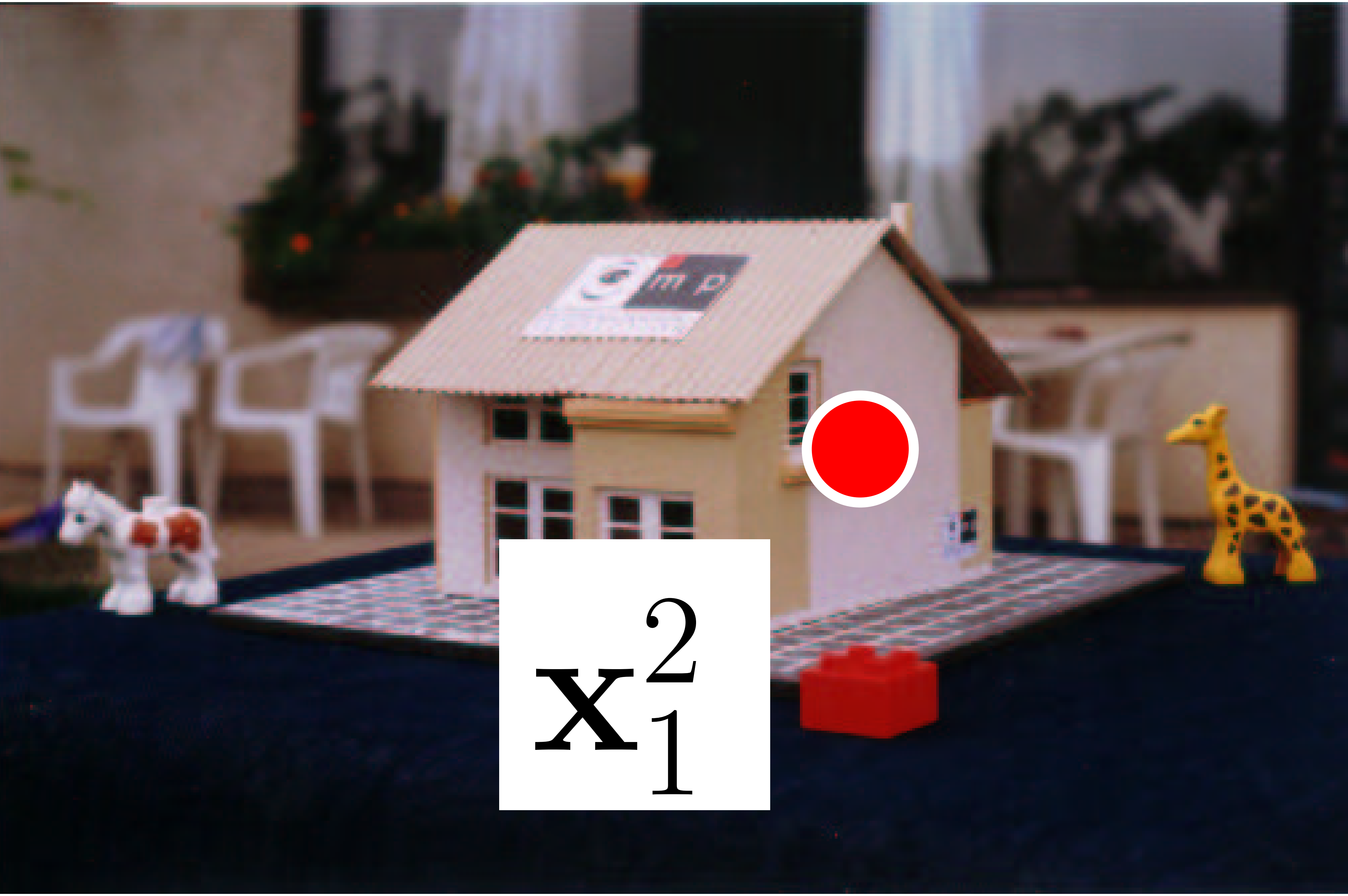
Repeat steps 1–4 until image points are sufficiently sampled. Image points with high number of votes are tentative inliers, other points are tentative outliers.

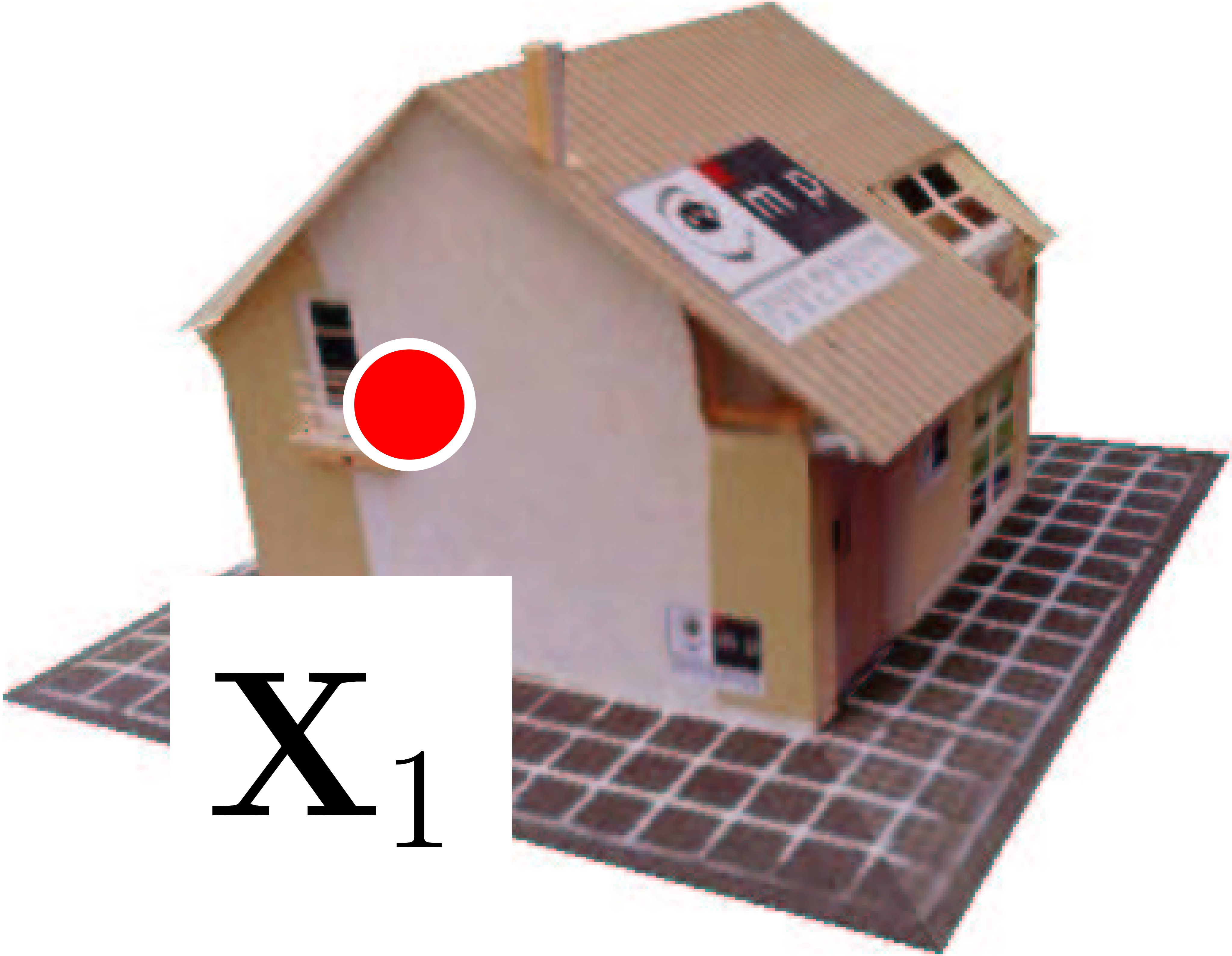
Projective Depths in Two Views – Geometrical Explanation

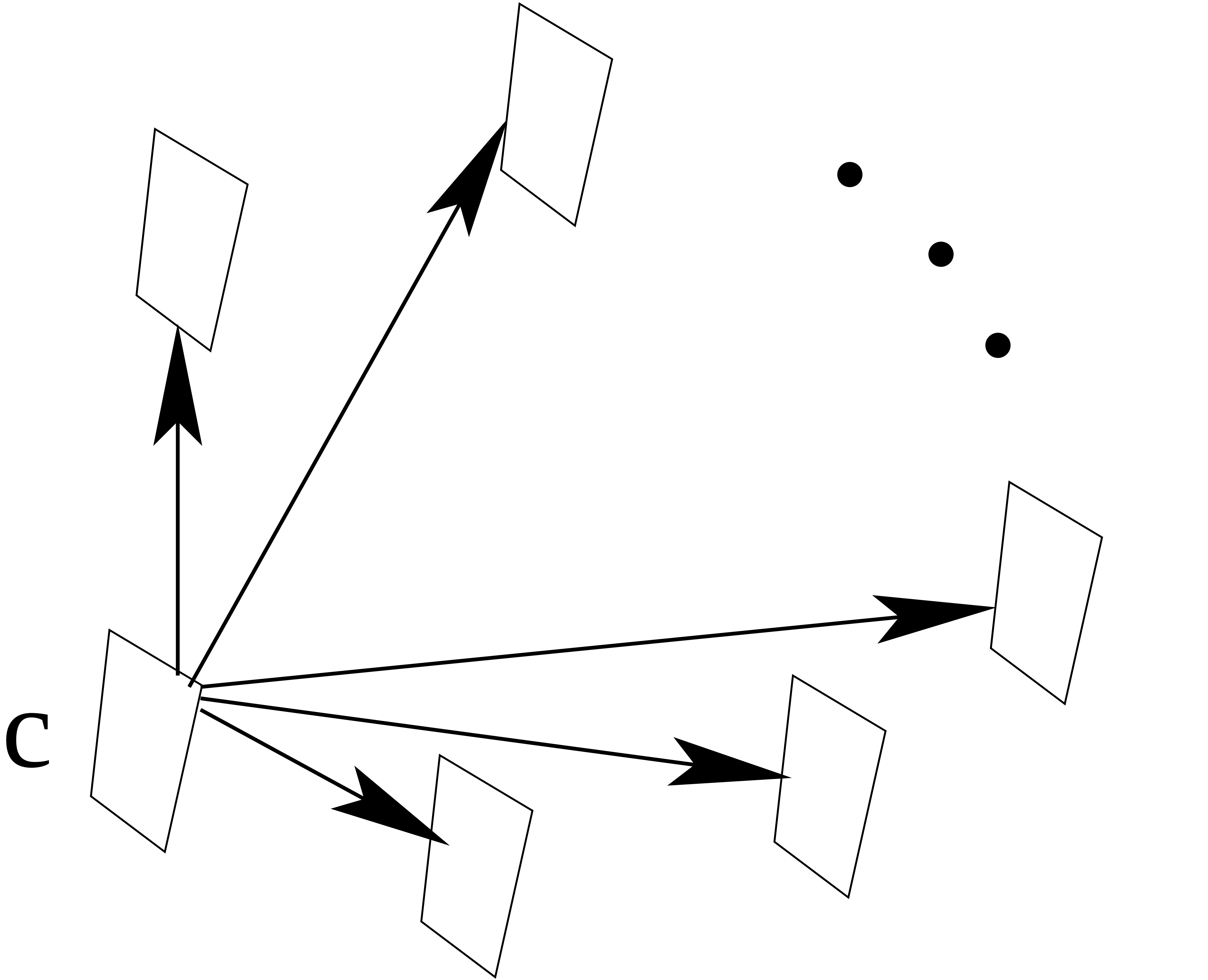


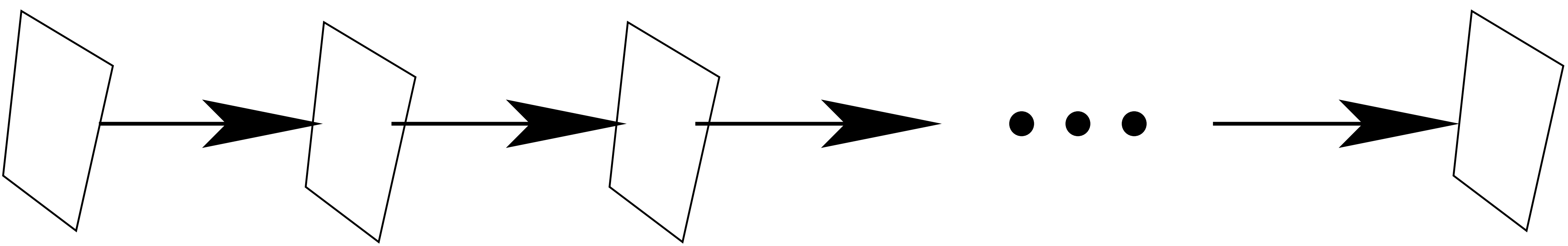
$$
\begin{aligned}
\lambda\mathbf{x}_\beta &= \tau\mathbf{e}_\beta + \lambda'\mathbf{x}'_\beta && / \ \mathbf{e}_\beta \wedge \\
\lambda(\mathbf{e}_\beta \wedge \mathbf{x}_\beta) &= \lambda'(\mathbf{e}_\beta \wedge \mathbf{x}'_\beta) \\
\mathbf{e}_\beta \wedge \mathbf{x}'_\beta &= [\mathbf{e}_\beta]_\times \mathtt{A}^{-1}\mathbf{x}'_{\beta'} = \alpha(\mathtt{F}\mathbf{x}'_{\beta'}) \\
\lambda(\mathbf{e}_\beta \wedge \mathbf{x}_\beta) &= \lambda'\alpha(\mathtt{F}\mathbf{x}'_{\beta'})
\end{aligned}
$$


$\mathbf{x}_1^1$


$\mathbf{x}_1^2$


$\mathbf{X}_1$


C

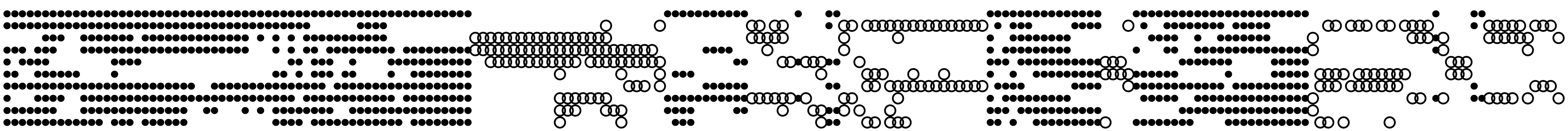


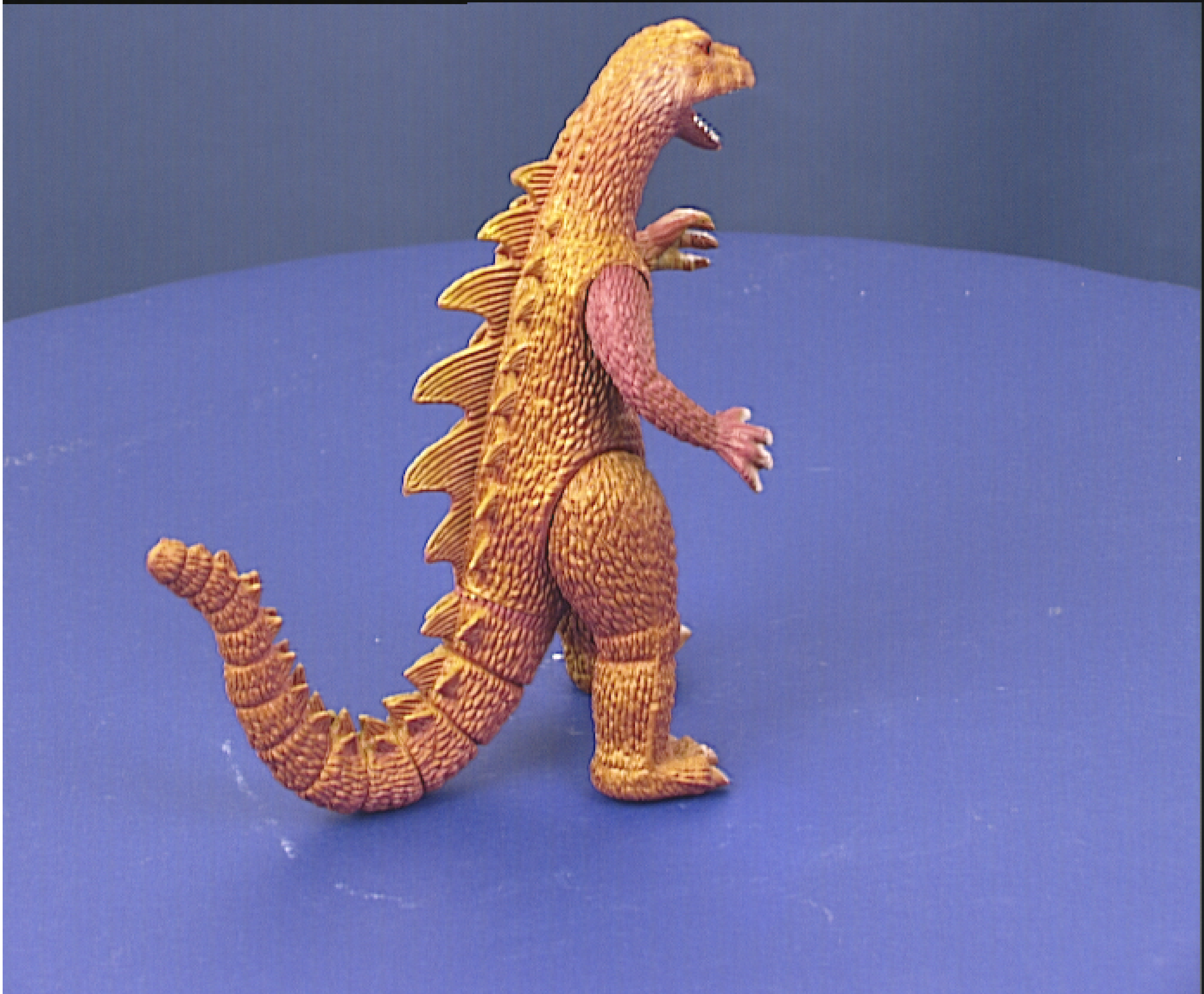
a 4-tuple of “LI” columns

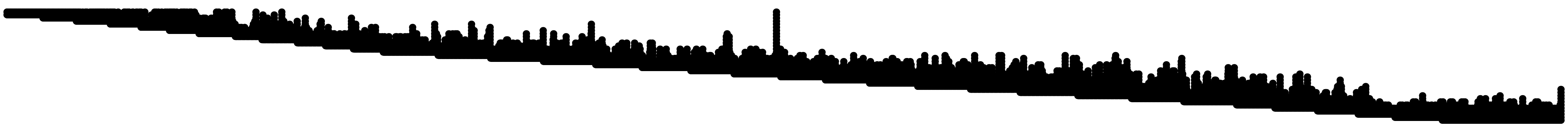
$$
\mathsf{B}_k = \begin{bmatrix} {\color{blue}?}\ \mathbf{x}_1^1 & \lambda_2^1\mathbf{x}_2^1 & \lambda_3^1\mathbf{x}_3^1 & \lambda_4^1\mathbf{x}_4^1 \\ \lambda_1^2\mathbf{x}_1^2 & \lambda_2^2\mathbf{x}_2^2 & \lambda_3^2\mathbf{x}_3^2 & \lambda_4^2\mathbf{x}_4^2 \\ \vdots & & & \\ \lambda_1^m\mathbf{x}_1^m & \lambda_2^m\mathbf{x}_2^m & \lambda_3^m\mathbf{x}_3^m & {\color{red}\times} \end{bmatrix} = \begin{bmatrix} {\color{blue}?}\ x_1^1 & \lambda_2^1 x_2^1 & \lambda_3^1 x_3^1 & \lambda_4^1 x_4^1 \\ {\color{blue}?}\ y_1^1 & \lambda_2^1 y_2^1 & \lambda_3^1 y_3^1 & \lambda_4^1 y_4^1 \\ {\color{blue}?}\ w_1^1 & \lambda_2^1 w_2^1 & \lambda_3^1 w_3^1 & \lambda_4^1 w_4^1 \\ \lambda_1^2 x_1^2 & \lambda_2^2 x_2^2 & \lambda_3^2 x_3^2 & \lambda_4^2 x_4^2 \\ \lambda_1^2 y_1^2 & \lambda_2^2 y_2^2 & \lambda_3^2 y_3^2 & \lambda_4^2 y_4^2 \\ \lambda_1^2 w_1^2 & \lambda_2^2 w_2^2 & \lambda_3^2 w_3^2 & \lambda_4^2 w_4^2 \\ \vdots & & & \\ \lambda_1^m x_1^m & \lambda_2^m x_2^m & \lambda_3^m x_3^m & {\color{red}\times} \\ \lambda_1^m y_1^m & \lambda_2^m y_2^m & \lambda_3^m y_3^m & {\color{red}\times} \\ \lambda_1^m w_1^m & \lambda_2^m w_2^m & \lambda_3^m w_3^m & {\color{red}\times} \end{bmatrix}
$$

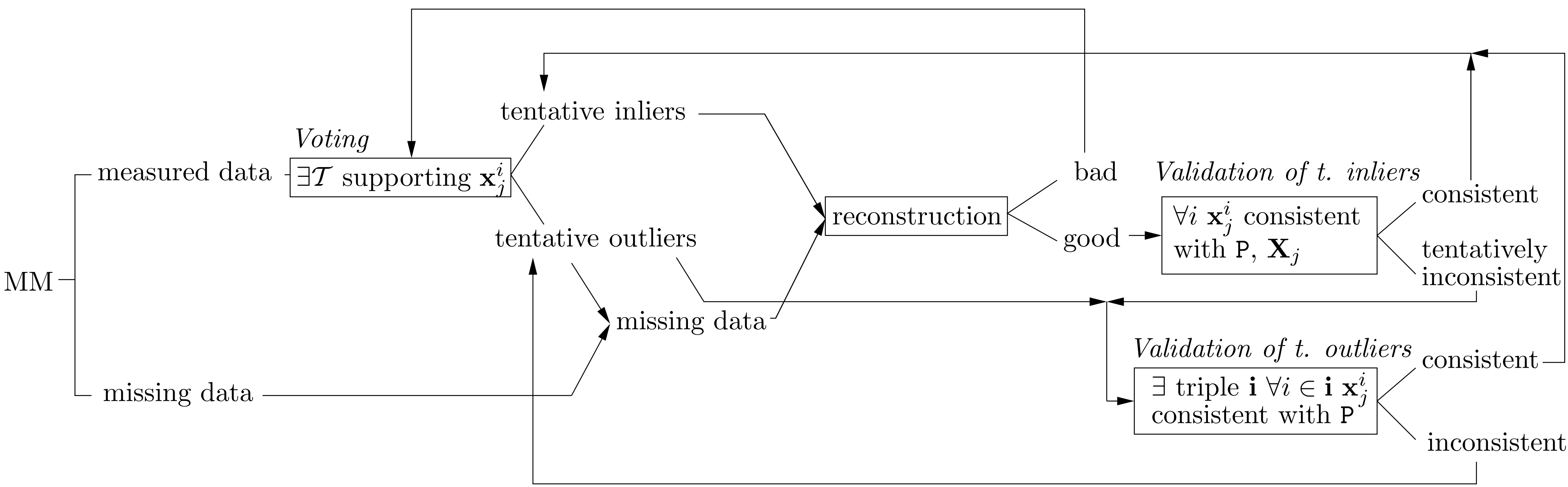
linear hull of all possible fillings

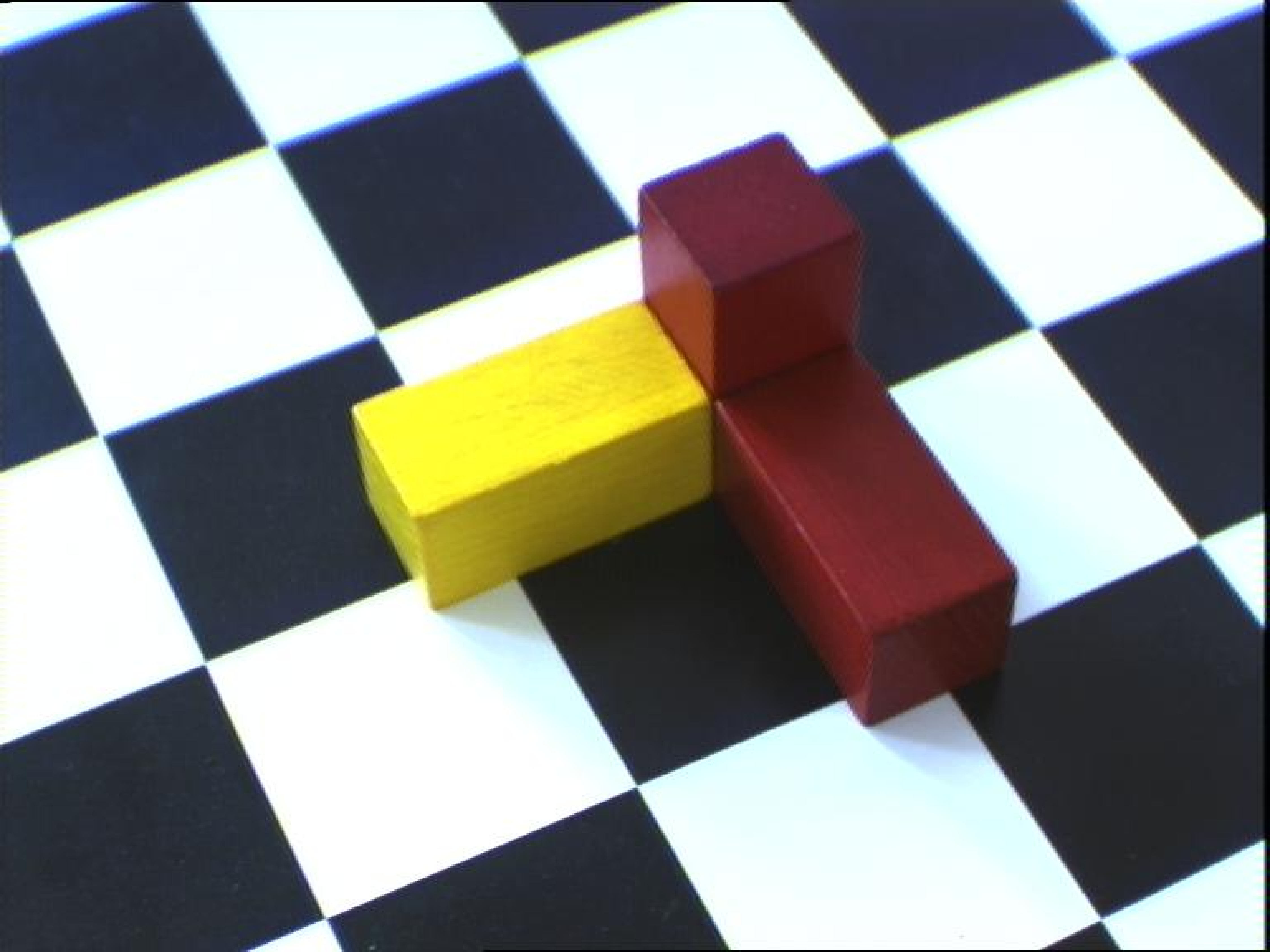
$$
\mathcal{B}_k \; = \; Span(\begin{bmatrix} {\color{blue}0} & x_1^1 & \lambda_2^1 x_2^1 & \lambda_3^1 x_3^1 & \lambda_4^1 x_4^1 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ {\color{blue}0} & y_1^1 & \lambda_2^1 y_2^1 & \lambda_3^1 y_3^1 & \lambda_4^1 y_4^1 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ {\color{blue}0} & w_1^1 & \lambda_2^1 w_2^1 & \lambda_3^1 w_3^1 & \lambda_4^1 w_4^1 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ \lambda_1^2 x_1^2 & {\color{blue}0} & \lambda_2^2 x_2^2 & \lambda_3^2 x_3^2 & \lambda_4^2 x_4^2 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ \lambda_1^2 y_1^2 & {\color{blue}0} & \lambda_2^2 y_2^2 & \lambda_3^2 y_3^2 & \lambda_4^2 y_4^2 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ \lambda_1^2 w_1^2 & {\color{blue}0} & \lambda_2^2 w_2^2 & \lambda_3^2 w_3^2 & \lambda_4^2 w_4^2 & {\color{red}0} & {\color{red}0} & {\color{red}0} \\ \vdots & & & & & & & \\ \lambda_1^m x_1^m & {\color{blue}0} & \lambda_2^m x_2^m & \lambda_3^m x_3^m & {\color{red}0} & {\color{red}\mathbf{1}} & {\color{red}0} & {\color{red}0} \\ \lambda_1^m y_1^m & {\color{blue}0} & \lambda_2^m y_2^m & \lambda_3^m y_3^m & {\color{red}0} & {\color{red}0} & {\color{red}\mathbf{1}} & {\color{red}0} \\ \lambda_1^m w_1^m & {\color{blue}0} & \lambda_2^m w_2^m & \lambda_3^m w_3^m & {\color{red}0} & {\color{red}0} & {\color{red}0} & {\color{red}\mathbf{1}} \end{bmatrix})
$$

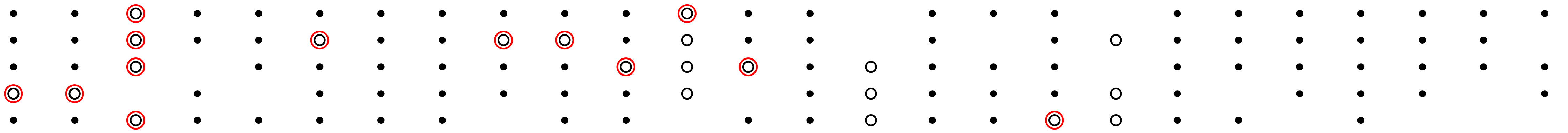


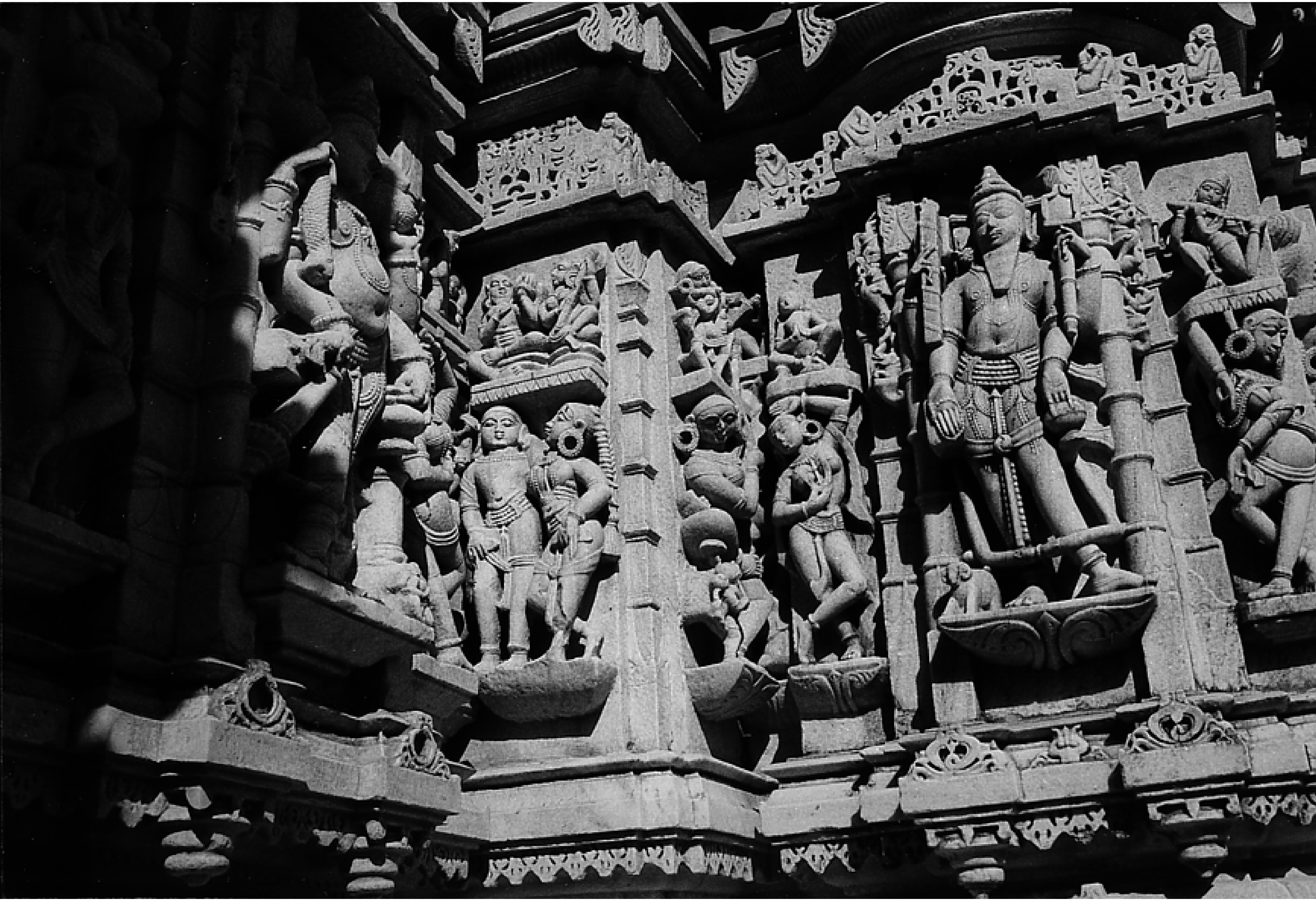


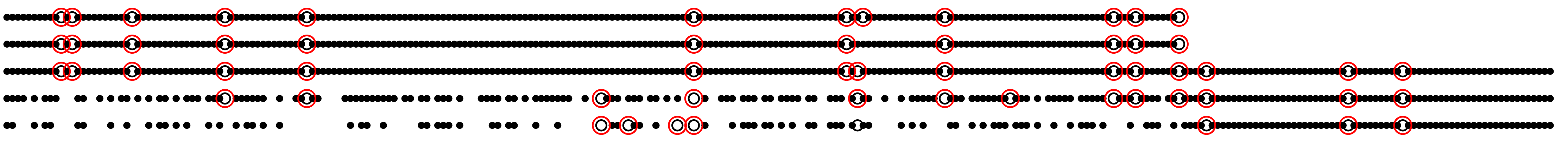


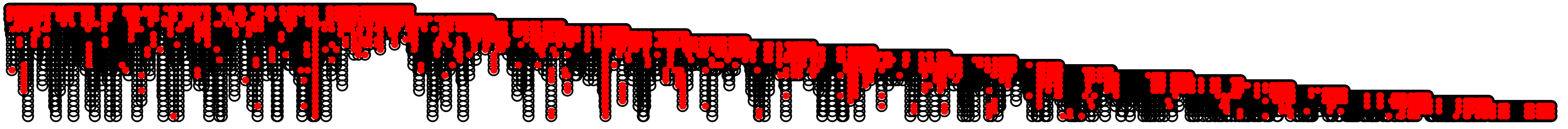

MM
measured data
missing data
Voting
$\exists\mathcal{T}$ supporting $\mathbf{x}_j^i$
tentative inliers
tentative outliers
missing data
reconstruction
bad
good
Validation of t. inliers
$\forall i$ $\mathbf{x}_j^i$ consistent with $\mathtt{P}$, $\mathbf{X}_j$
consistent
tentatively inconsistent
Validation of t. outliers
$\exists$ triple $\mathbf{i}$ $\forall i \in \mathbf{i}$ $\mathbf{x}_j^i$ consistent with $\mathtt{P}$
consistent
inconsistent

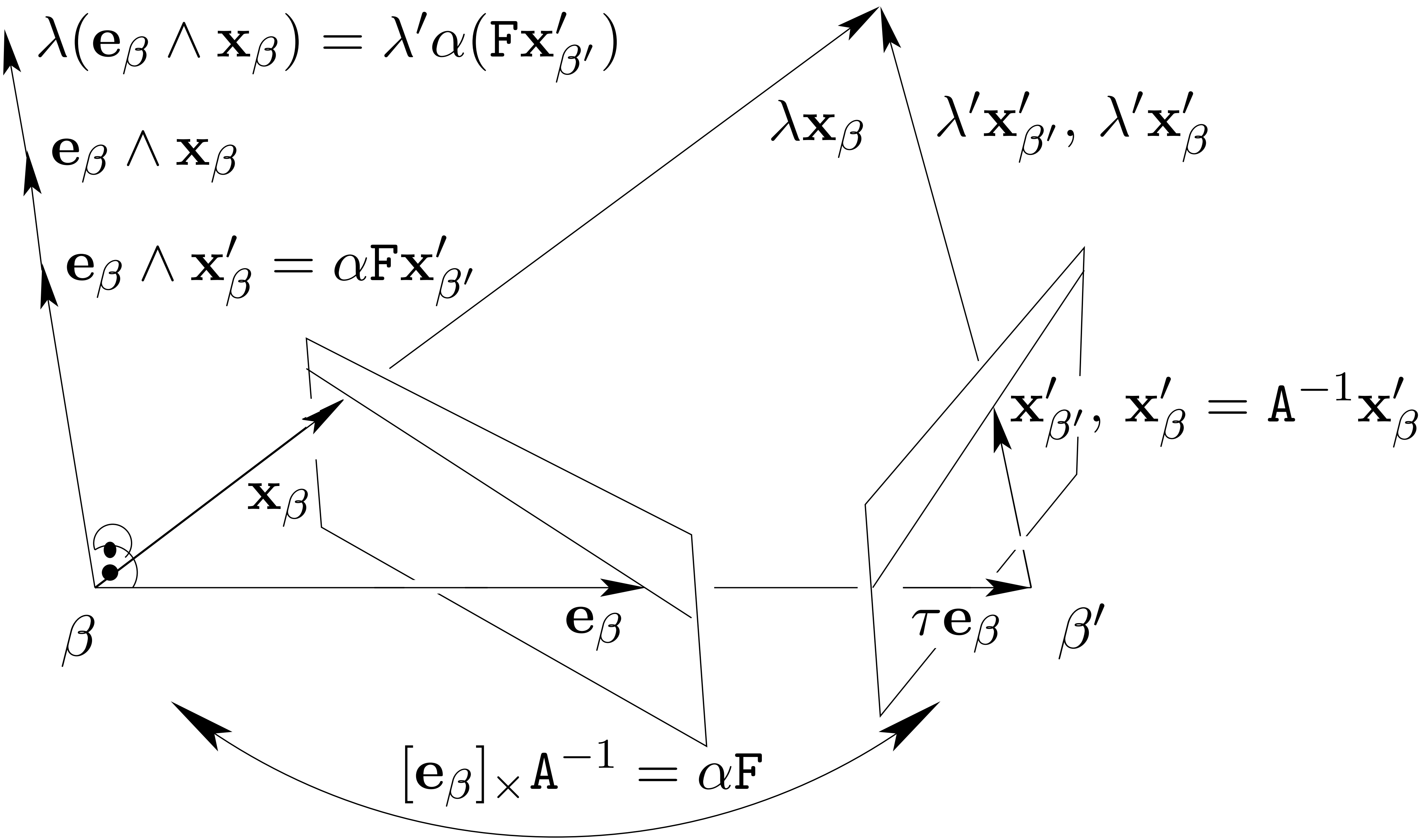












$\lambda(\mathbf{e}_\beta \wedge \mathbf{x}_\beta) = \lambda' \alpha(\mathtt{F}\mathbf{x}'_{\beta'})$
$\mathbf{e}_\beta \wedge \mathbf{x}_\beta$
$\mathbf{e}_\beta \wedge \mathbf{x}'_\beta = \alpha \mathtt{F}\mathbf{x}'_{\beta'}$
$\lambda \mathbf{x}_\beta$
$\lambda' \mathbf{x}'_{\beta'},\ \lambda' \mathbf{x}'_\beta$
$\mathbf{x}'_{\beta'},\ \mathbf{x}'_\beta = \mathtt{A}^{-1}\mathbf{x}'_{\beta'}$
$\mathbf{x}_\beta$
β
$\mathbf{e}_\beta$
$\tau \mathbf{e}_\beta$
β'
$[\mathbf{e}_\beta]_\times \mathtt{A}^{-1} = \alpha \mathtt{F}$