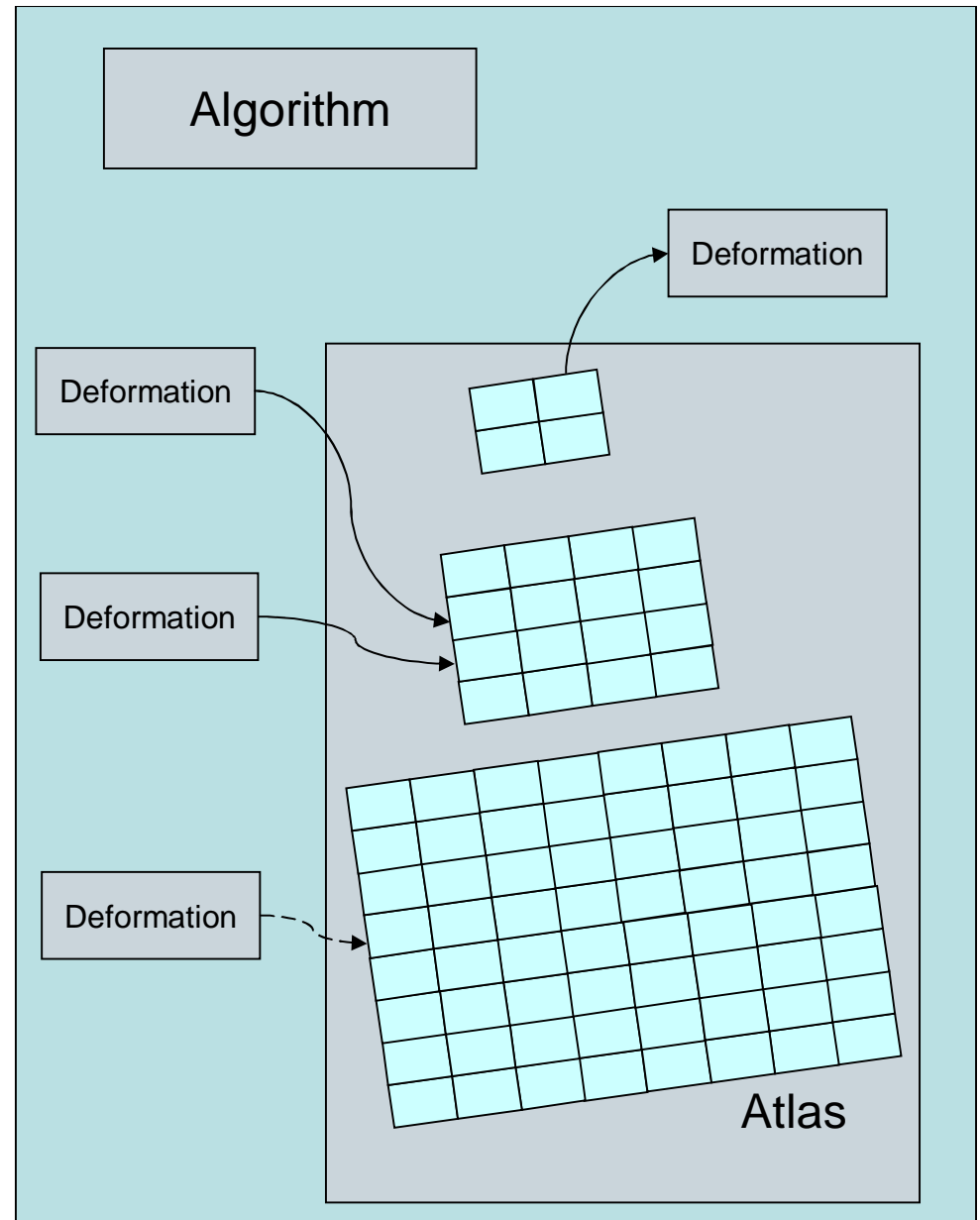


Map Alignment Calculation for TIGER

Simon Maskell,
Tom Cooper

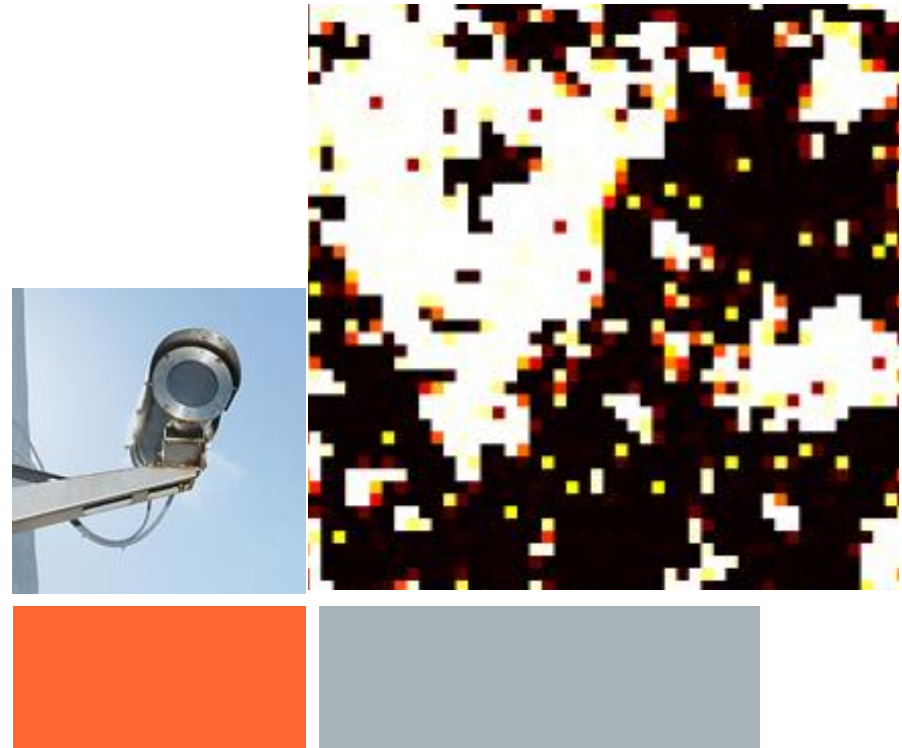
22 May 2009



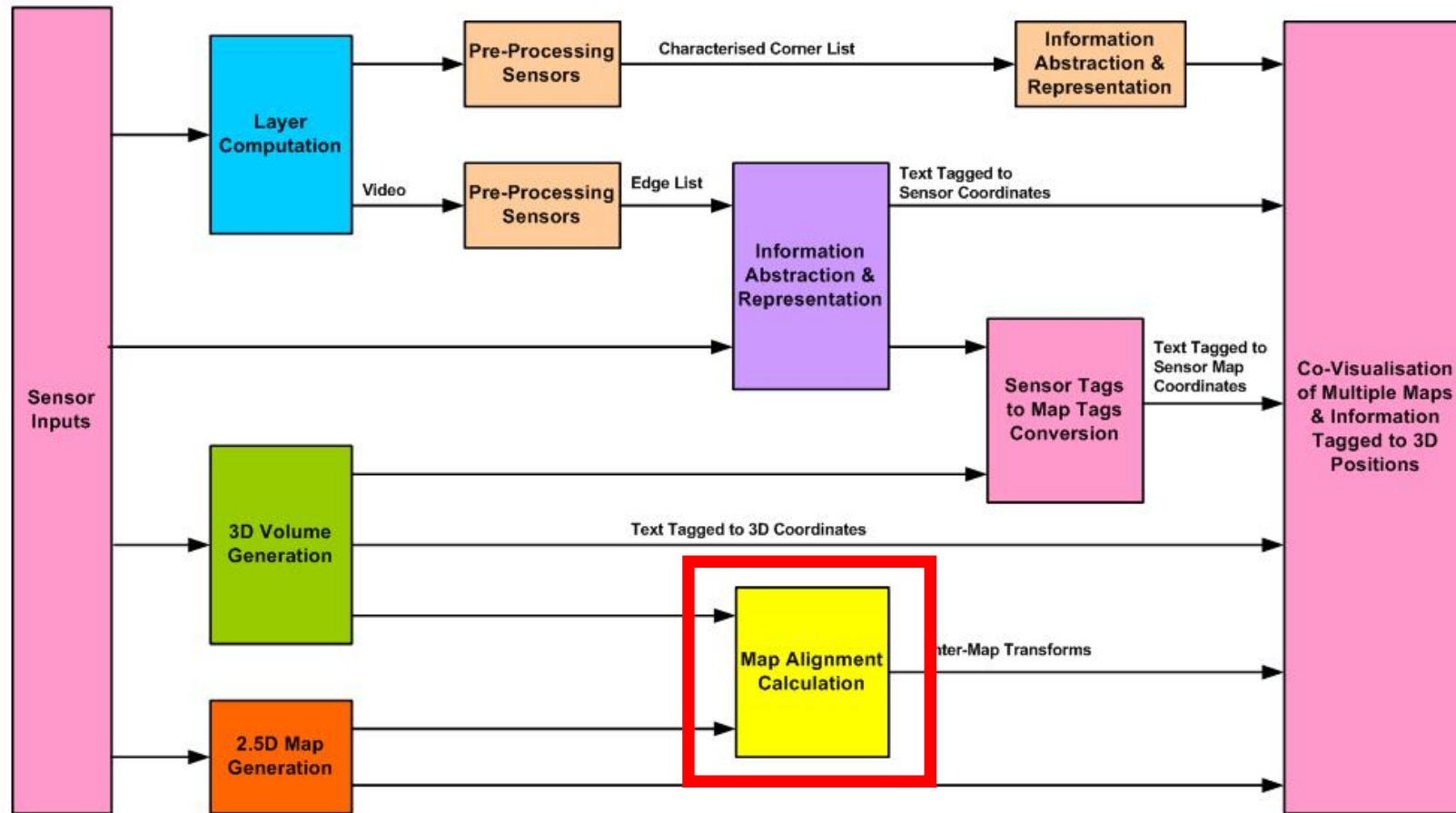
Contents

1. TIGER Context
2. Proposed Solution
3. Recommendations

- 01
TIGER Context

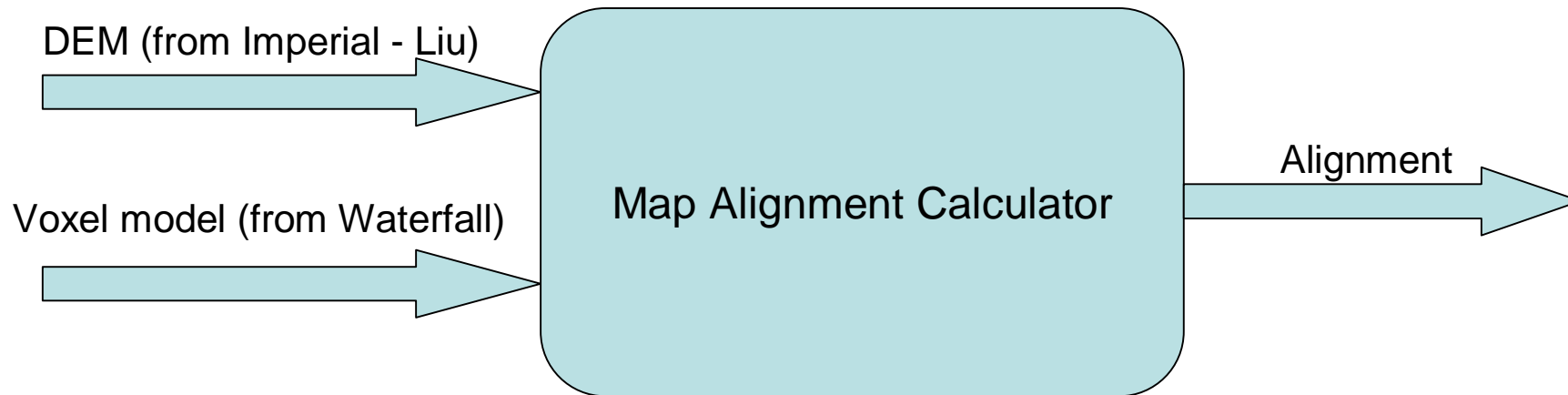


TIGER Context



- Imperial - Liu
- Imperial - Tania
- Imperial - Mike
- SELEX Galileo
- Waterfall
- QinetiQ
- Deep Vision

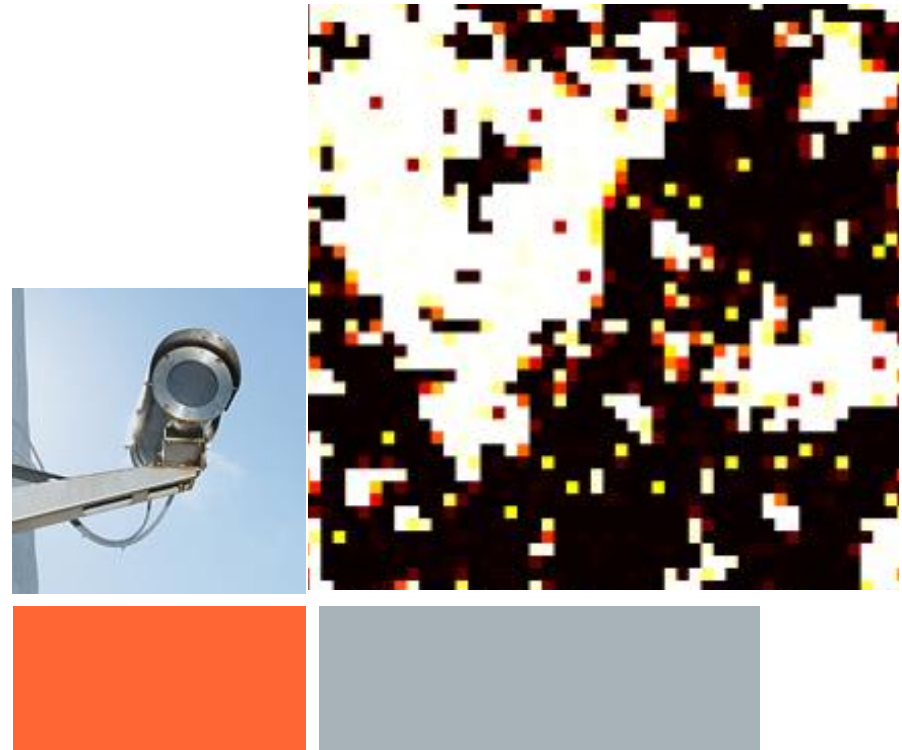
Detail of 'Map alignment calculation'



$$[X,Y,Z] = \text{DEMToVoxel}(\text{Alignment},x,y,z)$$

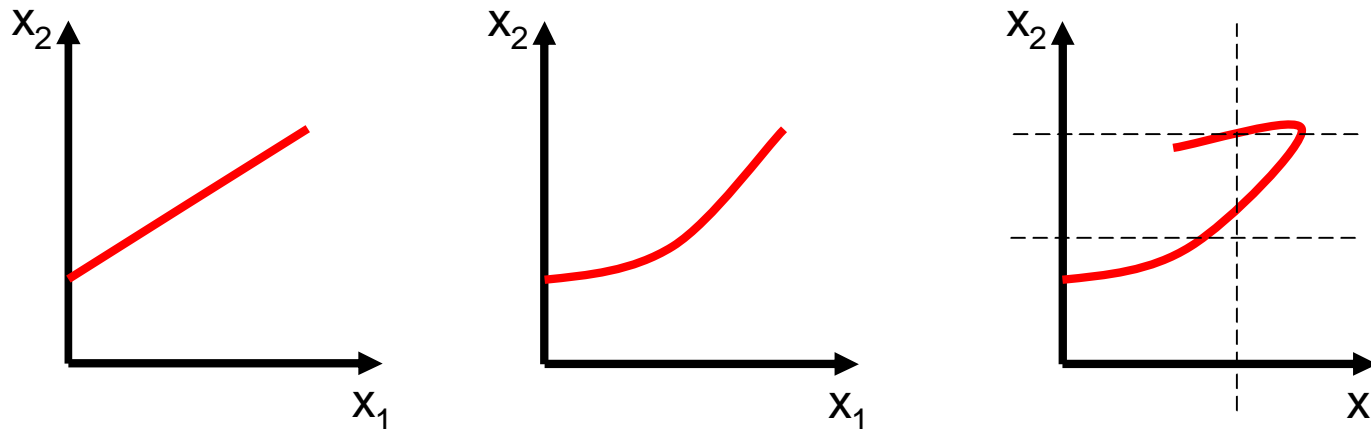
$$[x,y,z] = \text{VoxelToDEM}(\text{Alignment},X,Y,Z)$$

- 02
Proposed Solution



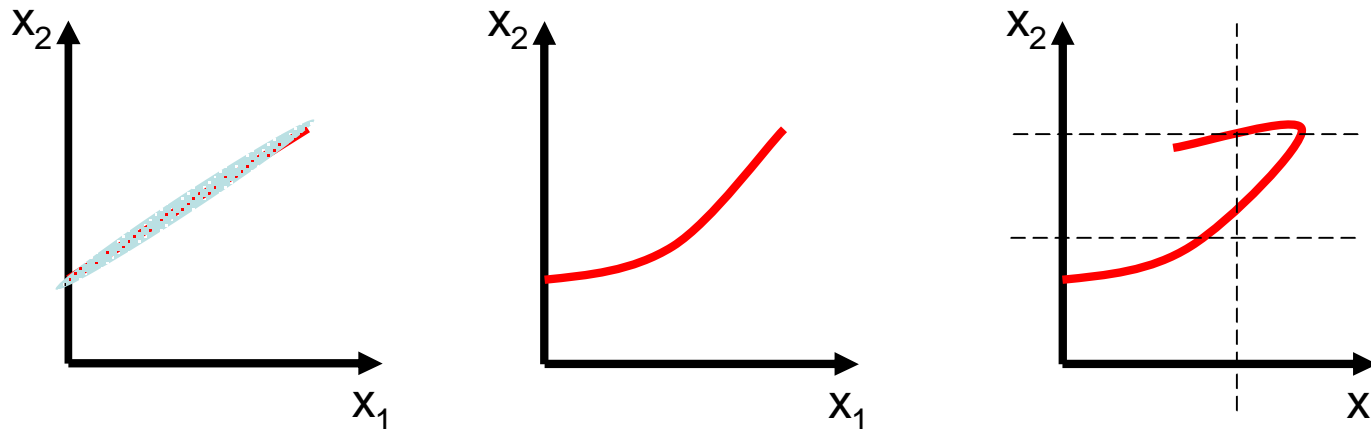
Modelling Transformations

- Transformations to consider:
 - Affine transformations (eg for a satellite image)
 - Projective transformations (eg for low-altitude UAV imagery)
 - Non-bijective transformations (eg from accumulation of INS errors)
- Solution: Pose transformation as regression with Gaussian Mixture Model



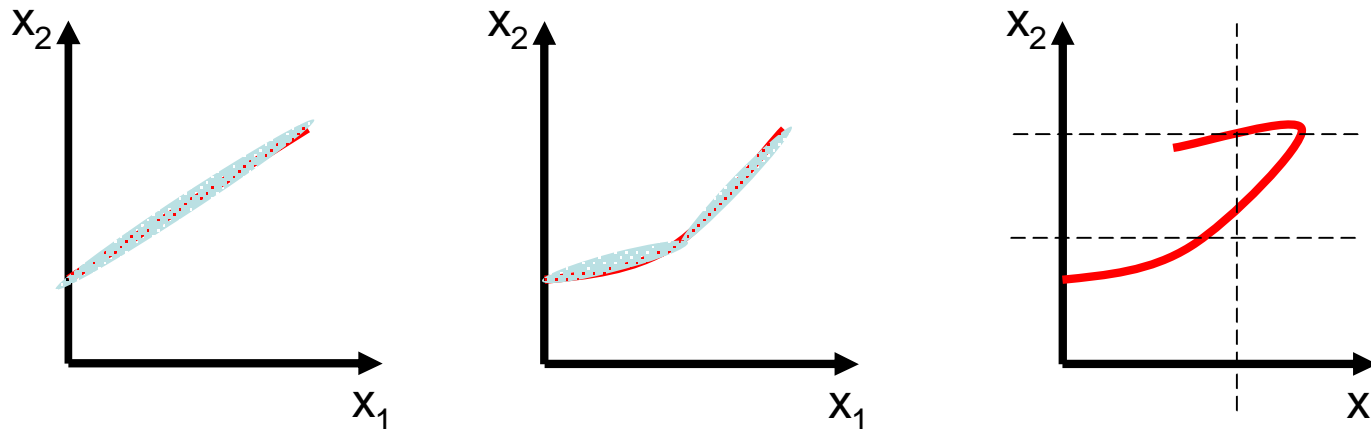
Modelling Transformations

- Transformations to consider:
 - Affine transformations (eg for a satellite image)
 - Projective transformations (eg for low-altitude UAV imagery)
 - Non-bijective transformations (eg from accumulation of INS errors)
- Solution: Pose transformation as regression with Gaussian Mixture Model



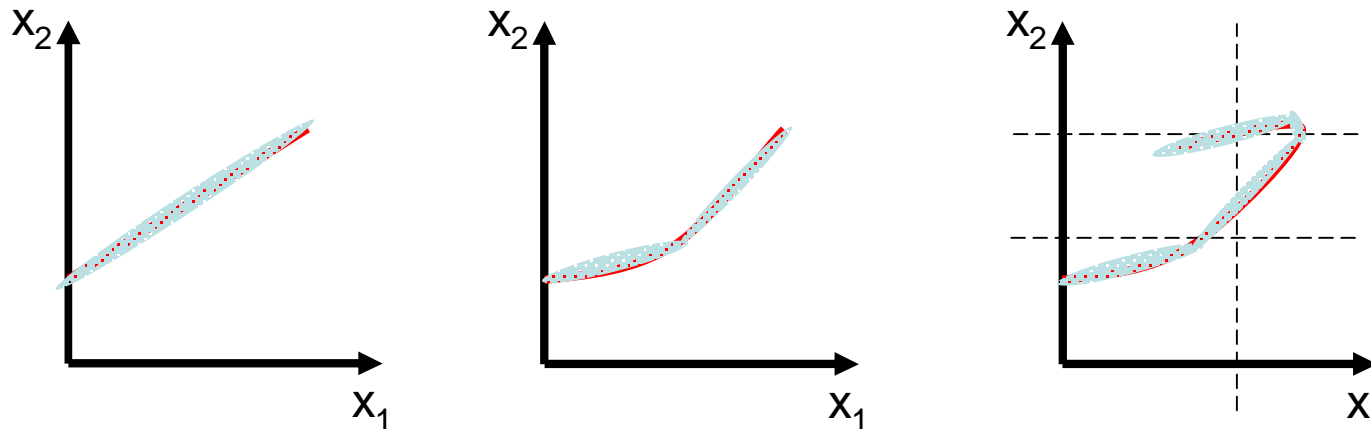
Modelling Transformations

- Transformations to consider:
 - Affine transformations (eg for a satellite image)
 - Projective transformations (eg for low-altitude UAV imagery)
 - Non-bijective transformations (eg from accumulation of INS errors)
- Solution: Pose transformation as regression with Gaussian Mixture Model



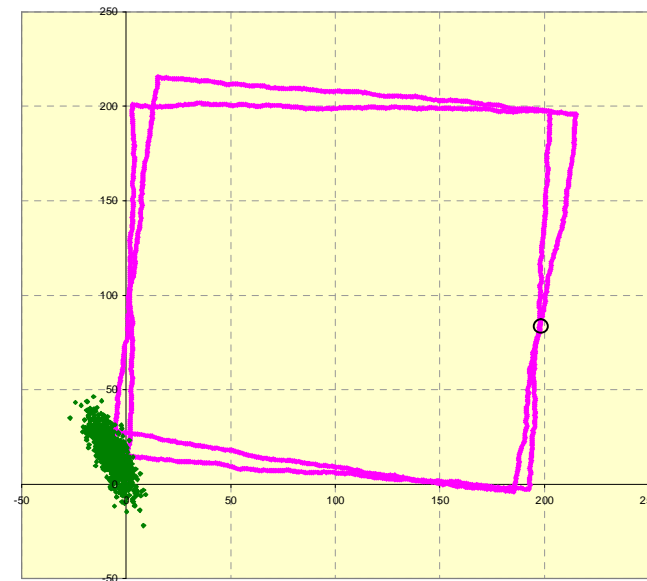
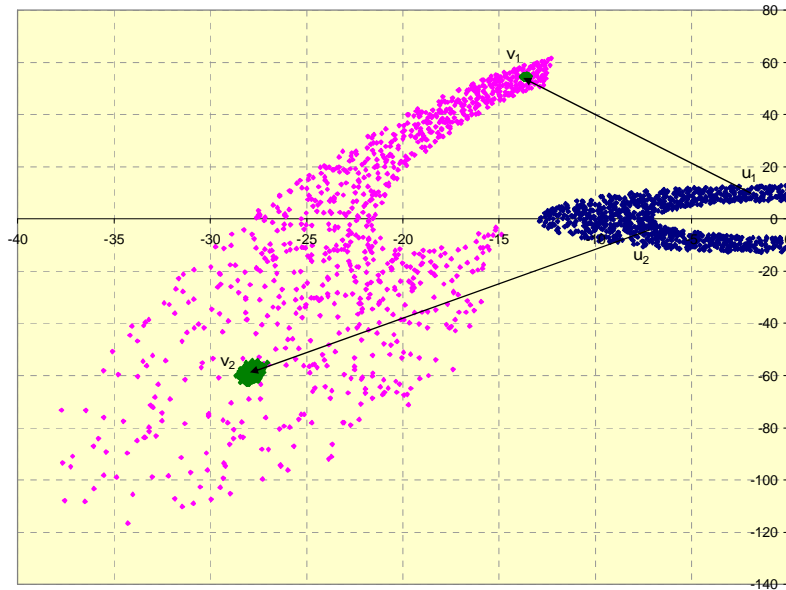
Modelling Transformations

- Transformations to consider:
 - Affine transformations (eg for a satellite image)
 - Projective transformations (eg for low-altitude UAV imagery)
 - Non-bijective transformations (eg from accumulation of INS errors)
- Solution: Pose transformation as regression with Gaussian Mixture Model

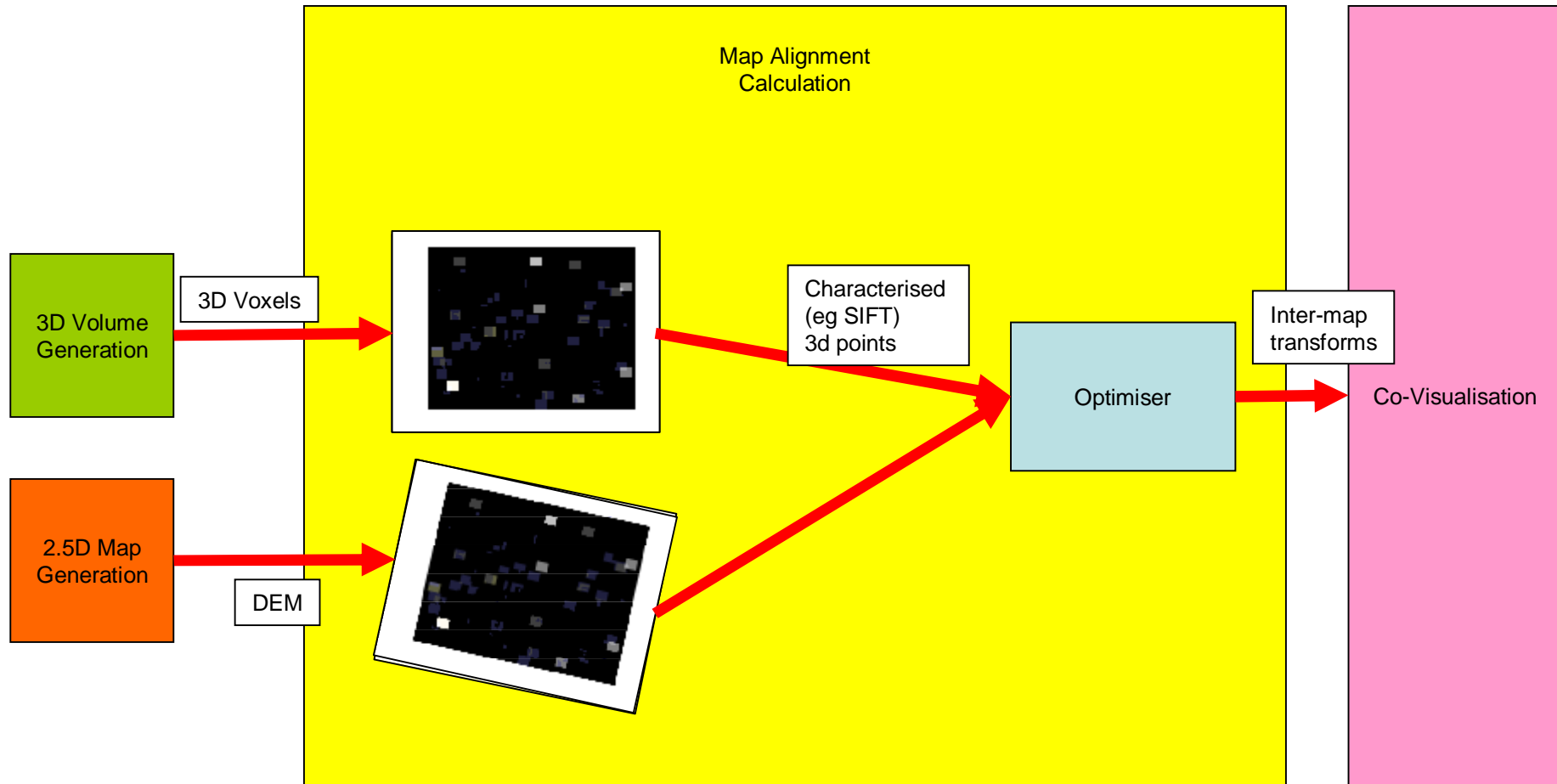


De-risking Gaussian mixture model for transformations

- Projectivity
- Simulated robot mapping
- 3D Transformations



Proposed Solution



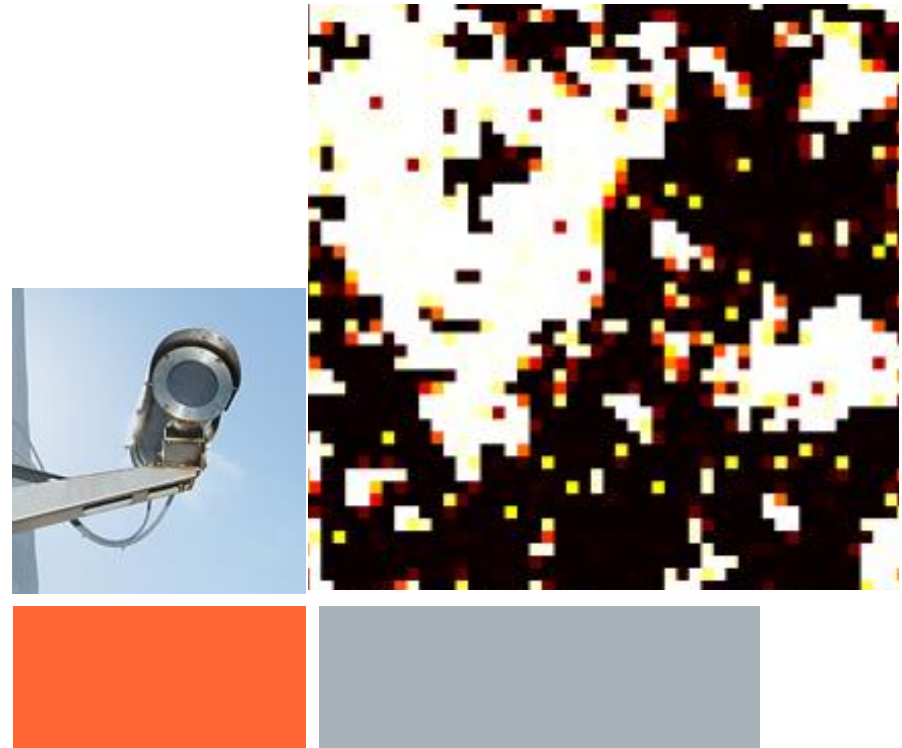
Plan of work

- Generate test data
 - Generate model world
 - Generate model DEM and voxel maps
 - Easy to reconfigure
 - To confirm system functionality
- Extract corners using visual processing toolbox (SIFT?)
 - Produces a set of tagged 3d points for each input
- Search for transformation
 - Define cost function (based on geometry of transformation, and match quality of corners)
 - Optimise using deterministic 'bundle adjustment'

Deterministic 'bundle adjustment'

- To find transformation 'θ' (and corner association 'A')
- Gibbs type scheme
 - 1. Adjust θ; 2. Adjust A; 3 Goto 1;
 - Adjust = sample => Gibbs sampling (slow)
 - Adjust = optimise => Iterative conditional optimisation (not multimodal)
- Better approach
 - Optimise θ subject to cost function $\int c(\theta, A) dA$
 - Maybe approximate by treating points independently
 - Works well if points are distinct
 - Maybe Efficient Hypothesis Management Algorithm (EHM/EHM2)

- 03
Recommendations



Recommendation

- Recommendation:
 - TIGER: Progress as planned towards demonstration