Imperial College London

Active Matching: Efficient Guided Search for Image Correspondence

Margarita Chli, Ankur Handa and Andrew Davison

http://www.doc.ic.ac.uk/~ajd/

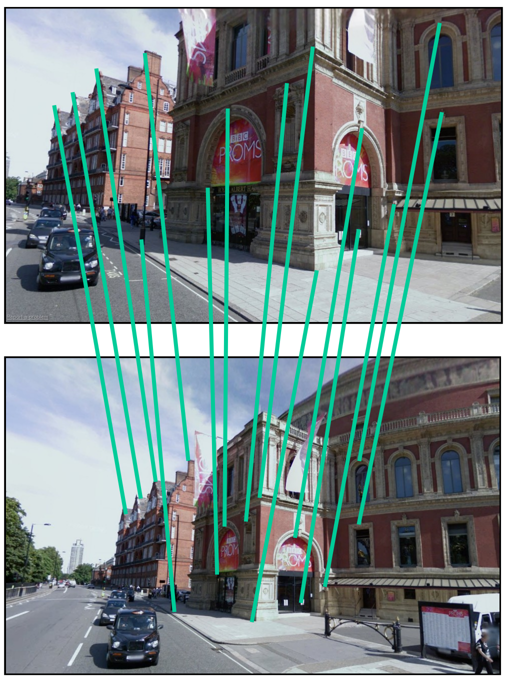
Robot Vision Group
Department of Computing, Imperial College London

Imperial College London

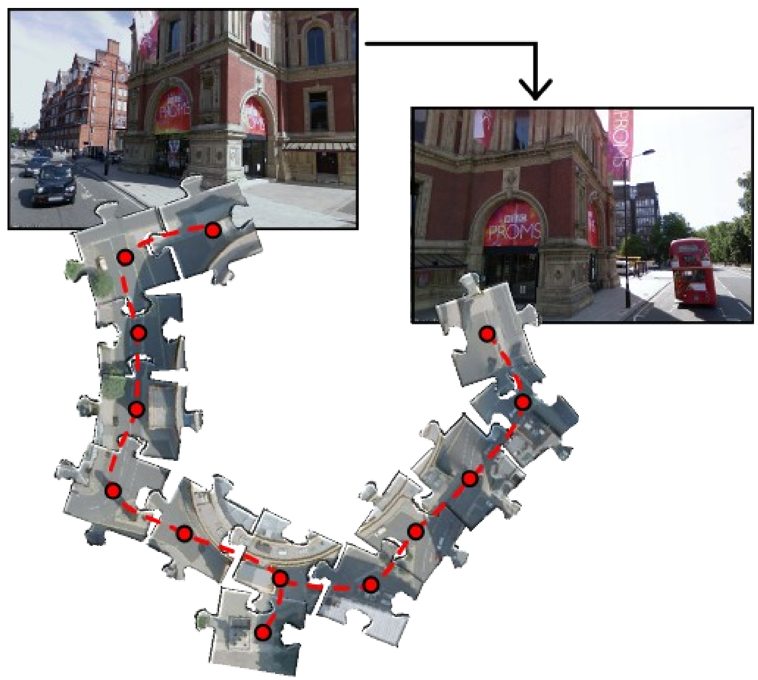
SLAM:
Simultaneous Localisation And Mapping

How can a body navigate in a previously unknown environment while constantly building a map of its workspace using on-board sensors only?

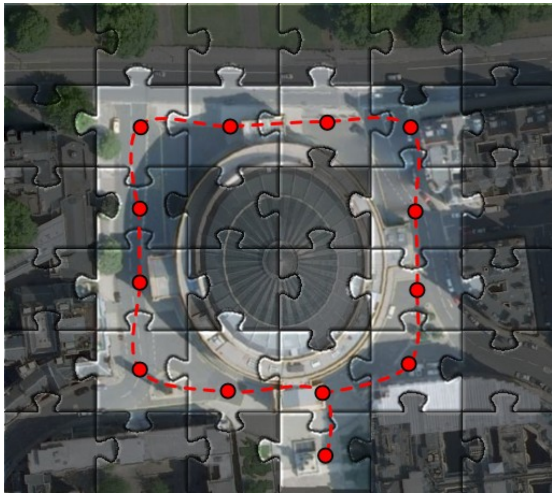
Components in a state-of-the-art system



Good local estimate of metric motion



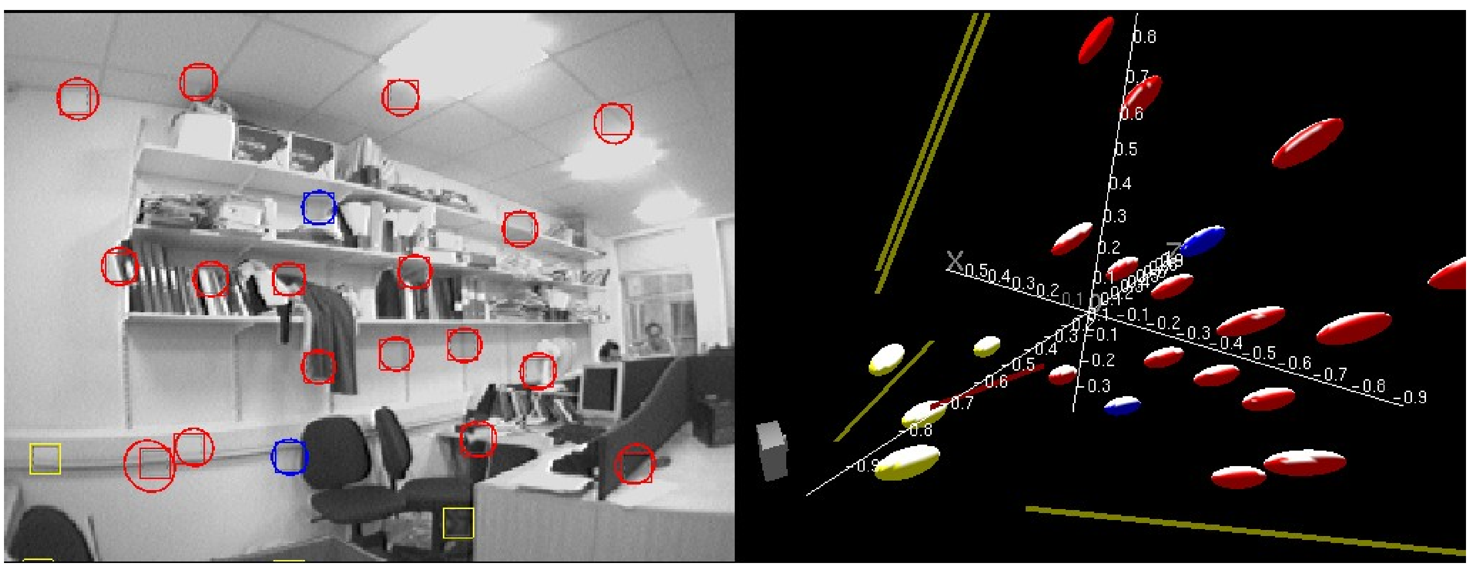
Mapping & loop-closure detection



Map management & optimisation

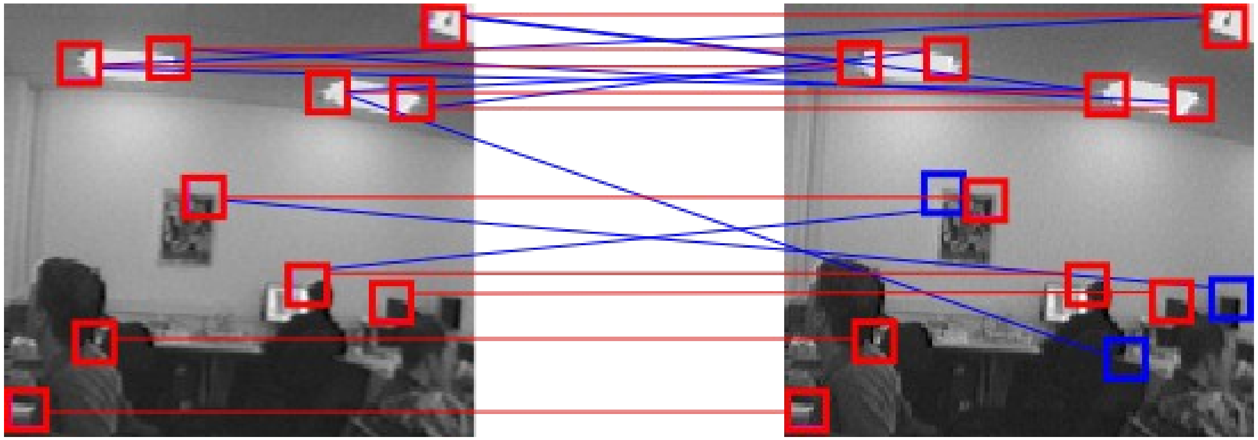
Matching = Data Association in images

- Find ***correspondence*** between known aspects of the world and image features
- Essential in tracking, recognition, SLAM, etc.
- Most generally, we want to match a ***probabilistic world model*** to the image.



Standard method: Get candidate matches & Resolve

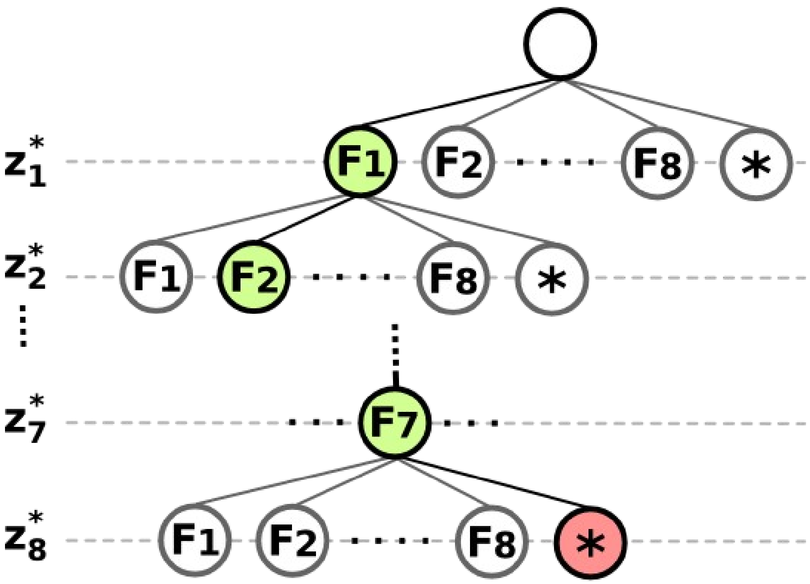
- First cue for matching: similar appearance of features



- Resolve inevitable **mismatches** by searching for consensus (agreement with global model)
- Example:
  RANSAC - randomly choose a set, hypothesize a solution & check and count

JCBB: Joint Compatibility Branch & Bound

- Interpretation tree evaluates joint probability of proposed matches **given probabilistic prior** on joint positions of features in image.



interpretation tree

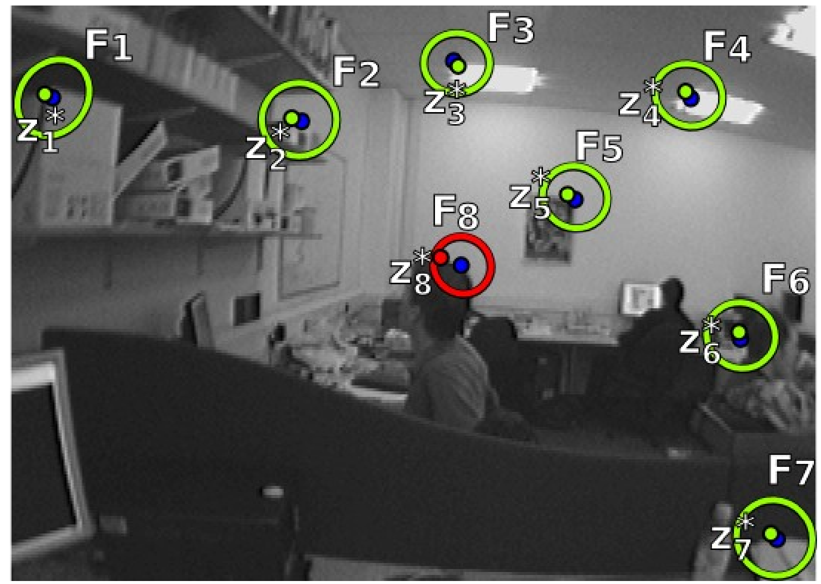


image space

- Still a '**detect first, resolve later**' method

[J.Neira and J.D.Tardós, IEEE Trans. Robotics and Automation, 2001]

Active Search: a more efficient sequential search

- **Single** multivariate Gaussian to describe the state of the camera and the feature measurements.
- Measure features **one-by-one** updating the joint Gaussian PDF.
- Order of measurement determined by **MI-scores** of candidate measurements with the camera state.
- Only toy examples; **cannot cope with ambiguity**

MEASURE FEATURE WITH HIGHEST MI-EFFICIENCY SCORE

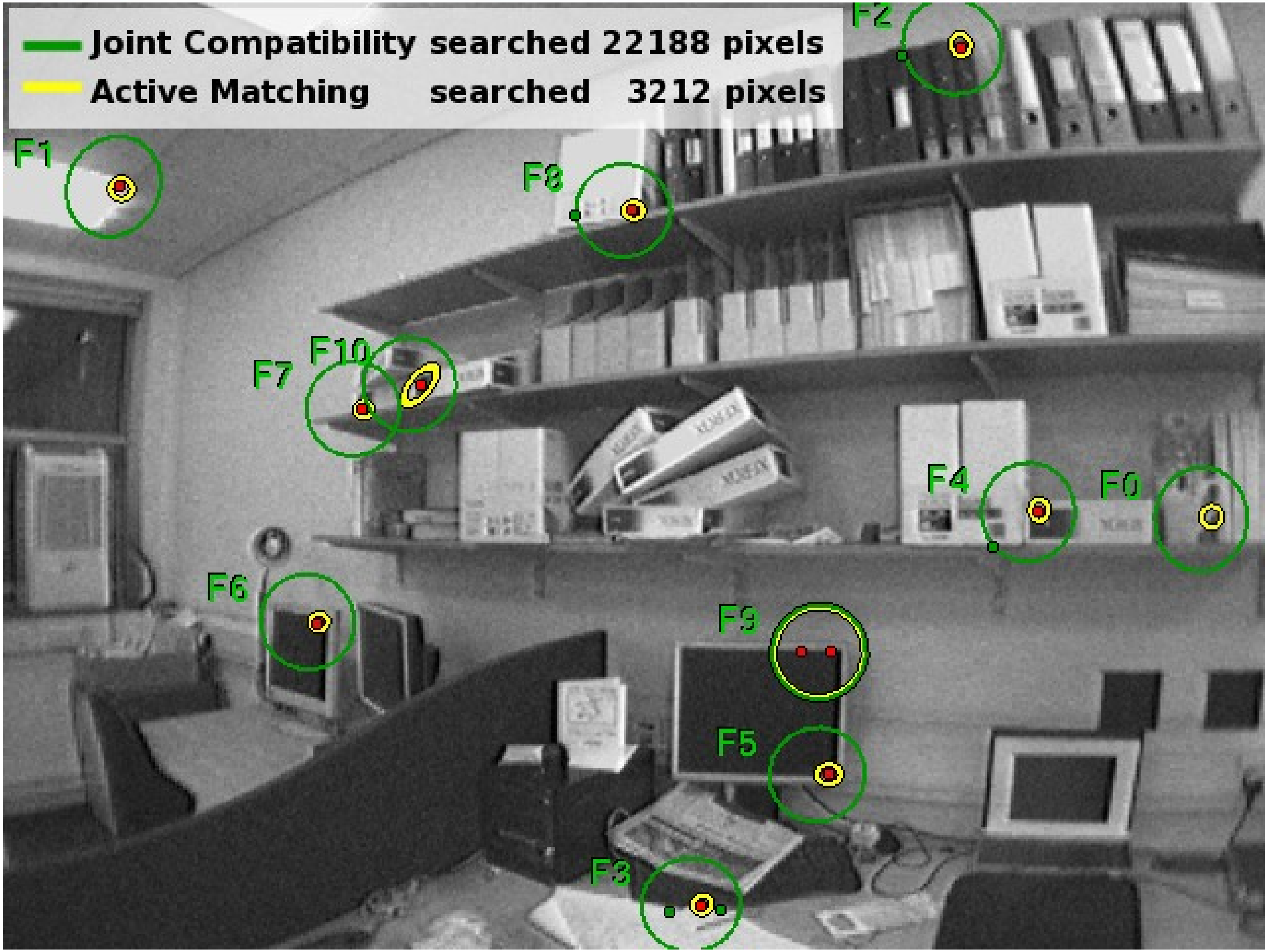
[Davison, ICCV 2005]

AM: Active Matching

- Step-by-step search for global consensus in the presence of ambiguity
- Uses a **Mixture of Gaussians** to describe the matching state
- **Mutual Information (MI) with matching state** defines where to look for matches next.
- In every matching step:
  - **PREDICT** which candidate gives max. MI-efficiency
  - **MEASURE** this candidate (search for a template match within a gated elliptical region)
  - **UPDATE** the mixture of Gaussians
- Define search completion in terms of **information theoretic** criterion.

[Chli, Davison ECCV 2008; Chli, Davison Robotics and Autonomous Systems 2009]

Active Matching: an example



AM: Search State

Stacked vector of all F features predicted to be visible:

$$\mathbf{x} = \begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \dots \\ \mathbf{z}_F \end{pmatrix}$$

e.g. in case of point features, then $\mathbf{z}_i$ holds the image coordinates for the corresponding feature

The PDF describing $\mathbf{x}$ is a mixture K Gaussians:

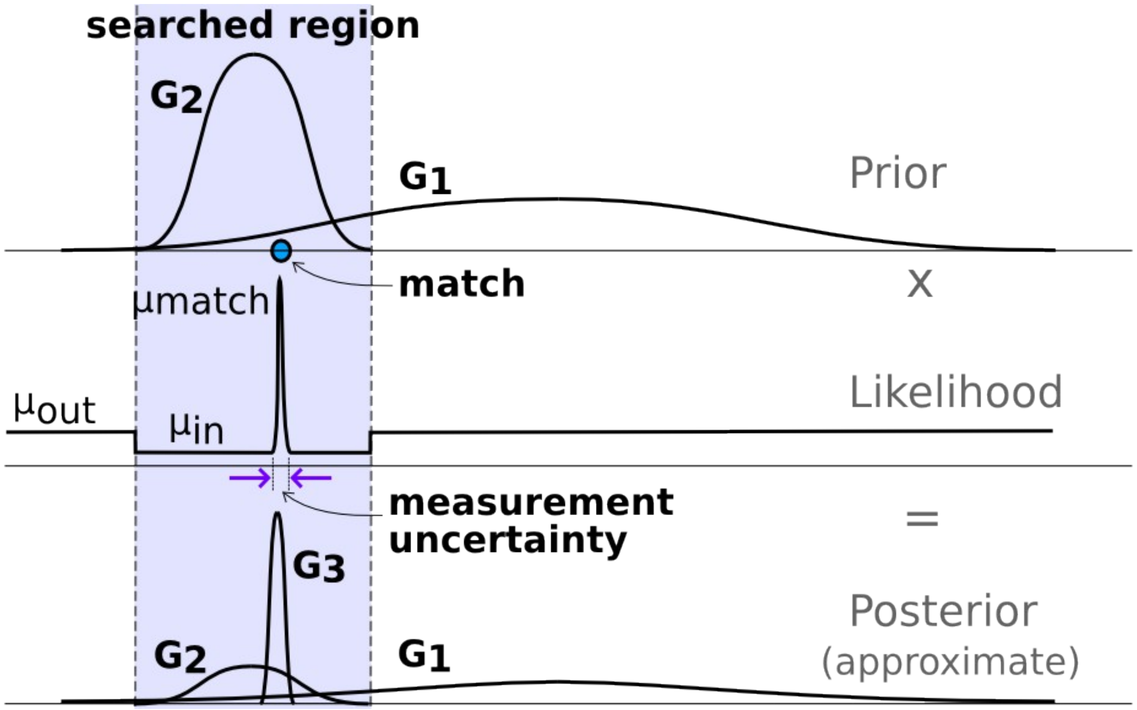
$$p(\mathbf{x}) = \sum_{i=1}^{K} \lambda_i \mathbf{G}_i$$

where $\mathbf{G}_i = \text{Gaussian}\{\hat{\mathbf{x}}_i, \text{S}_i\}$ and $\sum_{i=1}^{K} \lambda_i = 1$

Initialised with G_1 = search prior (e.g prediction coming from the application of a motion model) and $\lambda_1 = 1$.

AM: making a measurement

- Choose an ellipse to search from the K x F candidates (i.e. one {Feature, Gaussian} pair)
- Carry out exhaustive template matching in the region (ZNCC)
- Obtain $M = 0$ or more candidate matches above a threshold
- Perform Bayesian update of the mixture



Imperial College London

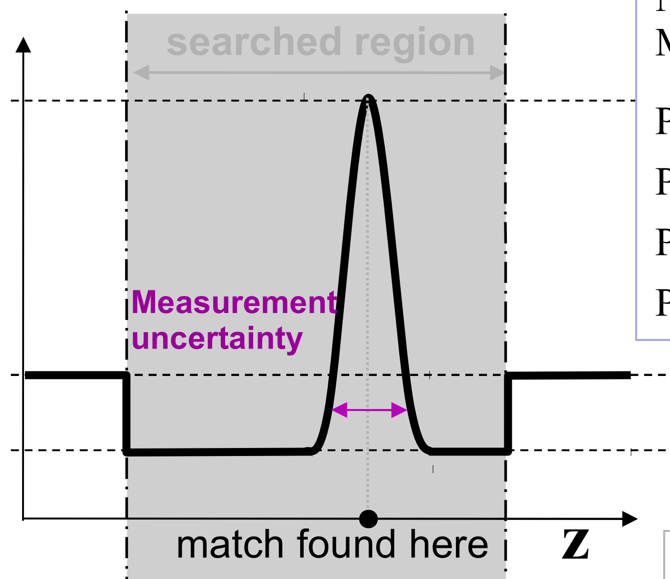
AM: Forming the Likelihood Function

- Likelihood: a mixture of discrete parts to account for ambiguous template matching

$$\mu_{match} = P_{tp} P_{fp}^{M-1} P_{tn}^{N-M}$$

$$\mu_{out} = P_{fp}^{M} P_{tn}^{N-M}$$

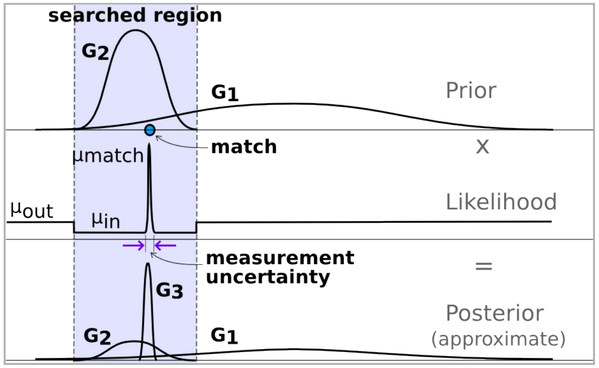
$$\mu_{in} = P_{fp}^{M} P_{fn} P_{tn}^{N-(M+1)}$$



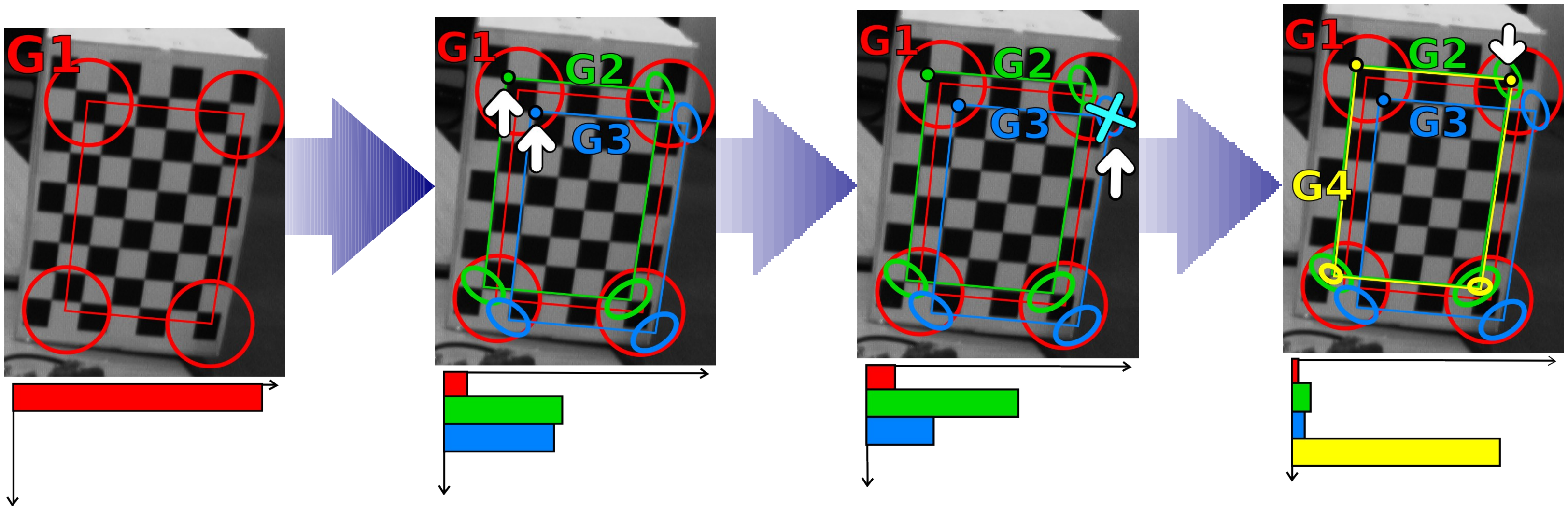
N: no. pixels searched
M: no. matches found

P_{tp}: prob. True Positives
P_{fp}: prob. False Positives
P_{tn}: prob. True Negatives
P_{fn}: prob. False Negatives

- Probabilities come from training



AM: Characteristics of an update



- **if** $M \geq 1$
  - A new, highly peaked Gaussian is spawned for each match to represent the hypothesis of match = True Positive
  - A new Gaussian gets higher weight if match is close to the centre of the search-ellipse (higher prior)

- **else if** $M = 0$
  - Does not kill the searched hypothesis immediately
  - Weight of searched Gaussian goes down & all other weights increase

- Prune weak Gaussians

AM: Order of measurement

- **Goal**: measure the ellipse that is predicted to reduce the overall uncertainty at the smallest image processing cost

$$\textbf{MI-efficiency of candidate measurement } z = \frac{\textbf{MI of } z}{\textbf{Area of } z\textbf{'s search-region}}$$

- To compute the MI of each z, we **predict** the shape of the mixture following potential measurement
(i.e. do a mock update for every candidate)

AM: Evaluating MI

$$\boxed{\mathbf{MI}(\mathbf{z}) = \mathbf{MI}_{\text{DISCRETE}}(\mathbf{z}) + \mathbf{MI}_{\text{CONTINUOUS}}(\mathbf{z})}$$

- Discrete part : RESOLUTION OF AMBIGUITY

Takes account of desire to move from many hypotheses with similar weights **towards a single, dominating hypothesis**

$$\begin{aligned}\mathbf{MI}_{\text{DISCRETE}}(\mathbf{z}) &= H(\mathbf{x}) - H(\mathbf{x}|\mathbf{z}) \\ &= H(\mathbf{x}) - P(\mathbf{z}=\text{null})H(\mathbf{x}|\mathbf{z}=\text{null}) - P(\mathbf{z}=\text{match})H(\mathbf{x}|\mathbf{z}=\text{match})\end{aligned}$$

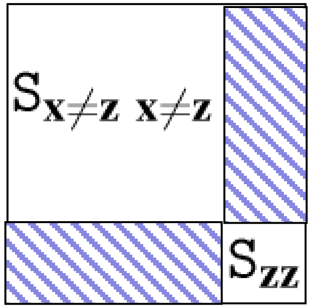
where the entropy of a mixture y of K Gaussians is $H(\mathbf{y}) = \sum_{i=1}^{K} \lambda_i \log_2 \frac{1}{\lambda_i}$

- Continuous part : IMPROVEMENT OF PRECISION

Takes account of desire to **reduce the variance** in newly spawned Gaussian

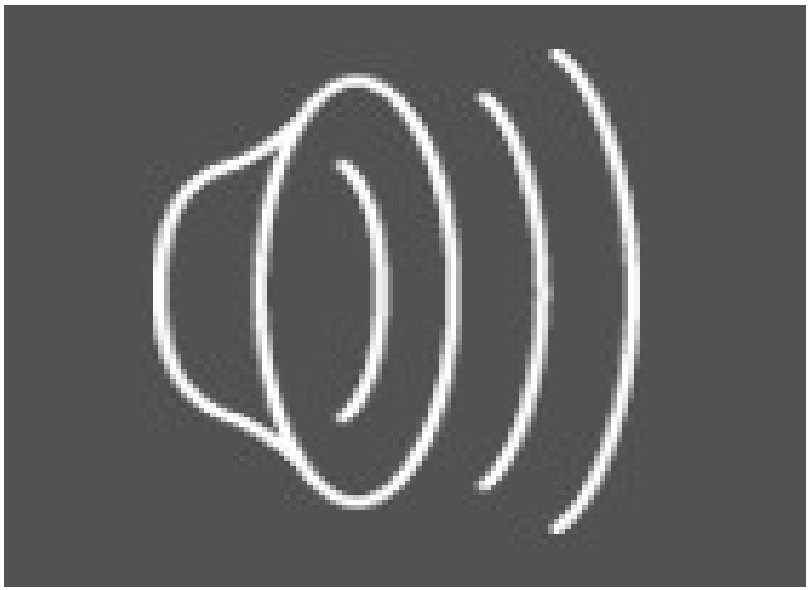
$$\mathbf{MI}_{\text{CONTINUOUS}}(\mathbf{z}) = \frac{1}{2} P(\mathbf{z}=\text{match}) \lambda_{\text{NEW}} \log_2 \frac{|\mathsf{S}_{\mathbf{zz}}||\mathsf{S}_{\mathbf{x}\neq\mathbf{z}\ \mathbf{x}\neq\mathbf{z}}|}{|\mathsf{S}|}$$

S =



Results on SLOW camera motion

Original sequence 30 Hz



Sub-sampled sequence to 15Hz

: AM search-regions

Standard search-regions:

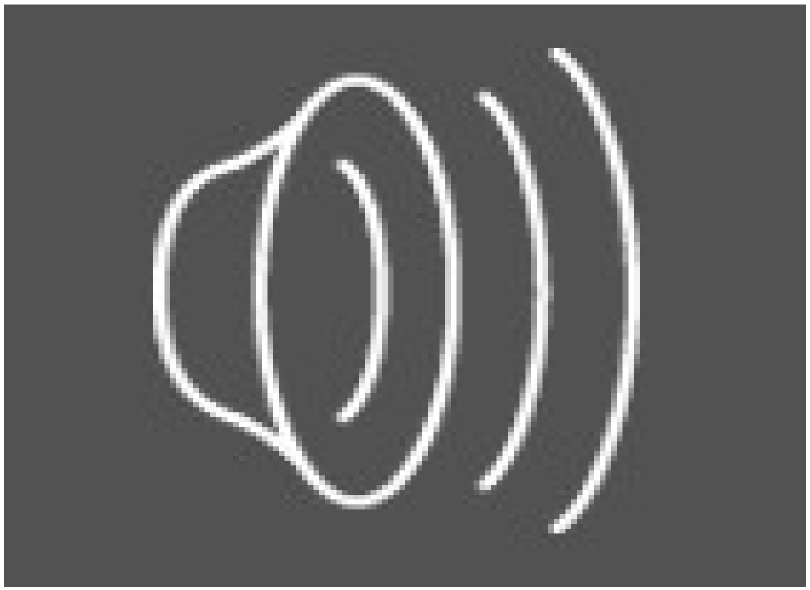
: successful search

: failed search

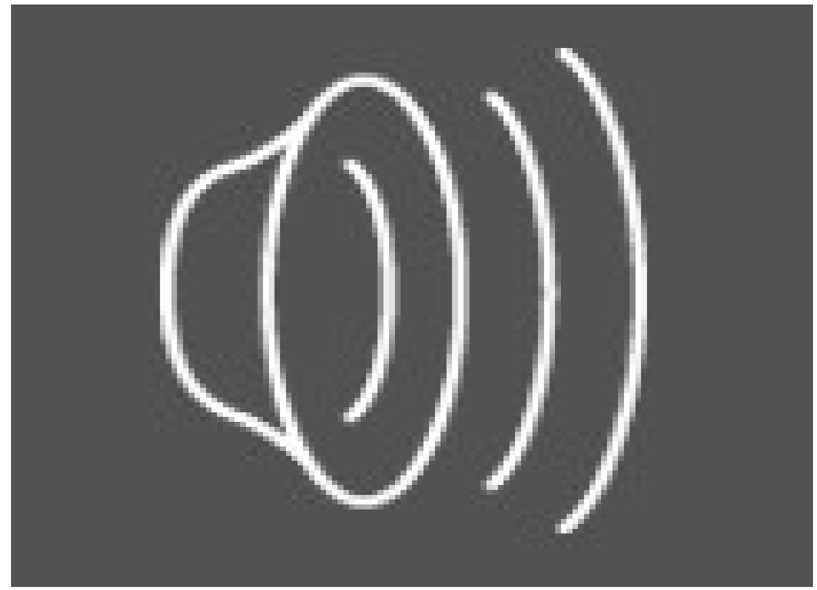
	One tracking step	Matching only	No. pixels searched [relative ratio]	Max no. live Gaussians
Slow Sequence at 30Hz (592 frames)				
JCBB	34.9ms	28.7ms	21517 [6.89:1]	-
AM	19.5ms	16.1ms	3124	5
Slow Sequence at 15Hz (296 frames)				
JCBB	59.4ms	52.4ms	40548 [7.78:1]	-
AM	22.0ms	15.6ms	5212	6

Results on FAST camera motion

Original sequence 30 Hz



Sub-sampled sequence to 15Hz



: AM search-regions

Standard search-regions:

: successful search

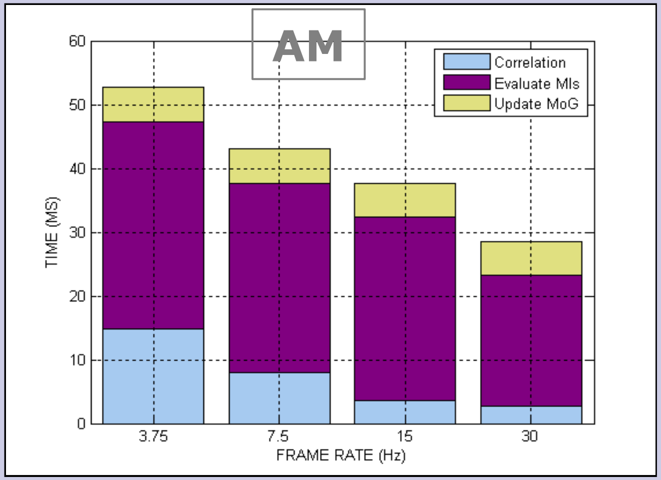
: failed search

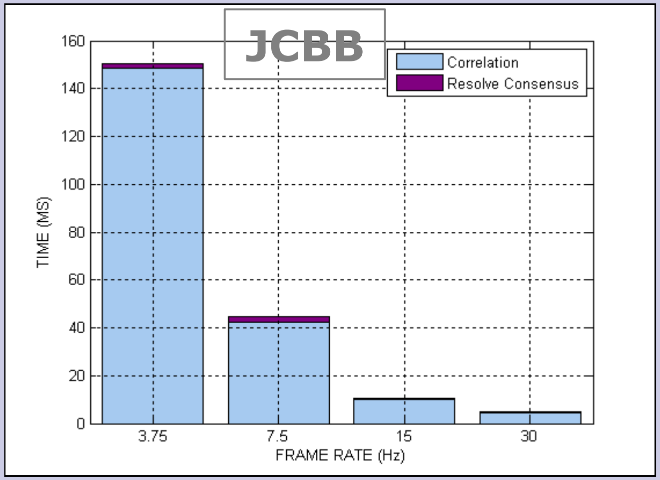
	One tracking step	Matching only	No. pixels searched [relative ratio]	Max no. live Gaussians
Fast Sequence at 30Hz (752 frames)				
JCBB	56.8ms	51.2ms	40341 [8.01:1]	-
AM	21.6ms	16.1ms	5039	7
Fast Sequence at 15Hz (376 frames)				
JCBB	102.6ms	97.1ms	78675 [8.27:1]	-
AM	38.1ms	30.4ms	9508	10

Imperial College London

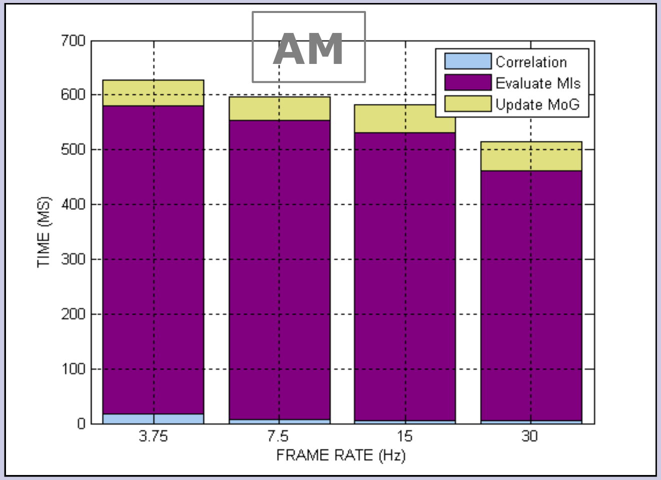
AM: Timings breakdown

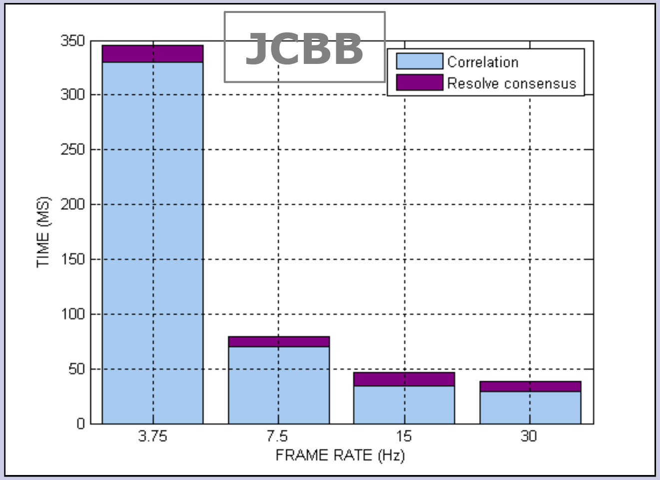
Matching 20 features per frame





Matching 40 features per frame

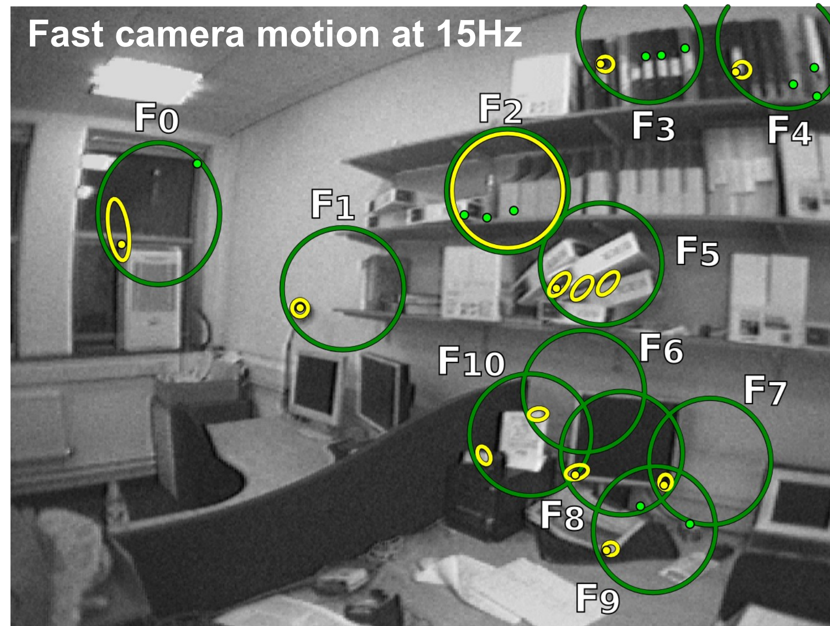




Imperial College London

AM: Conclusions

- ✓ Robust, multi-hypothesis fully probabilistic matching
- ✓ Matching results **agree** with JCBB
- ✓ Seamless integration of feature matching probability models
- ✓ **Real-time** operation for < 20 features
- ✓ **Reduction** by 8x in image proc. operations
  ⇨ especially beneficial at lower frame rates
- ✓ **Fewer mismatches** encountered
  = avoid confusing the matcher
- ✘ Very **expensive** for more than 20 features

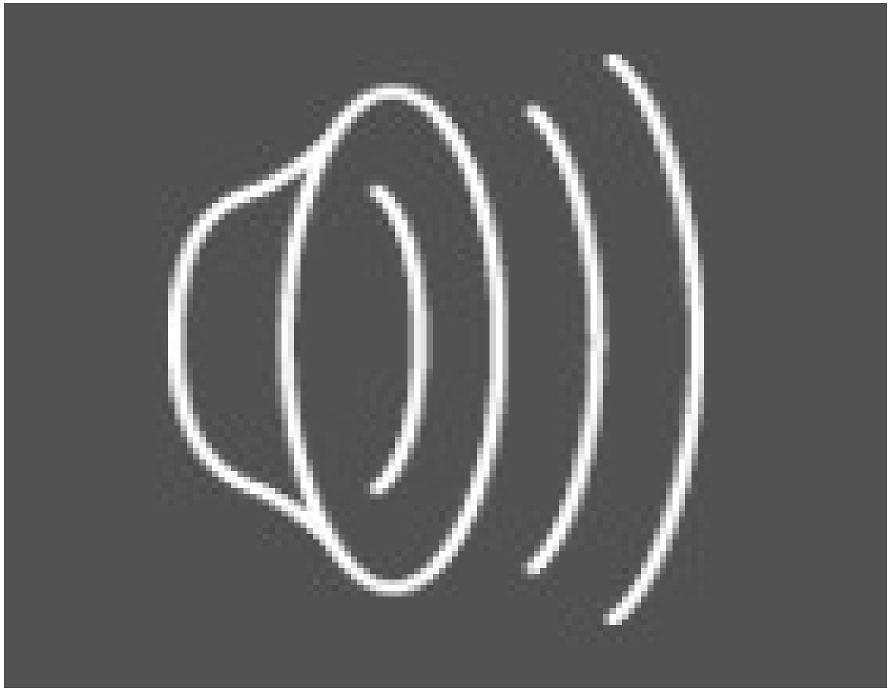


: AM regions **: matches encountered by AM**

: JCBB regions **: extra matches encountered with JCBB**

Imperial College London

The special character of Visual Maps



Usually: **regularly covisible** features are strongly correlated

But: not true if their motion is not **coherent** – they give different info about the camera motion.

We propose:

- A method to discover a **fully hierarchical** correlation structure using only image-based measurements of feature correlations during online SLAM filtering.

[Chli, Davison ICRA 2009]

Imperial College London

Pairwise MI

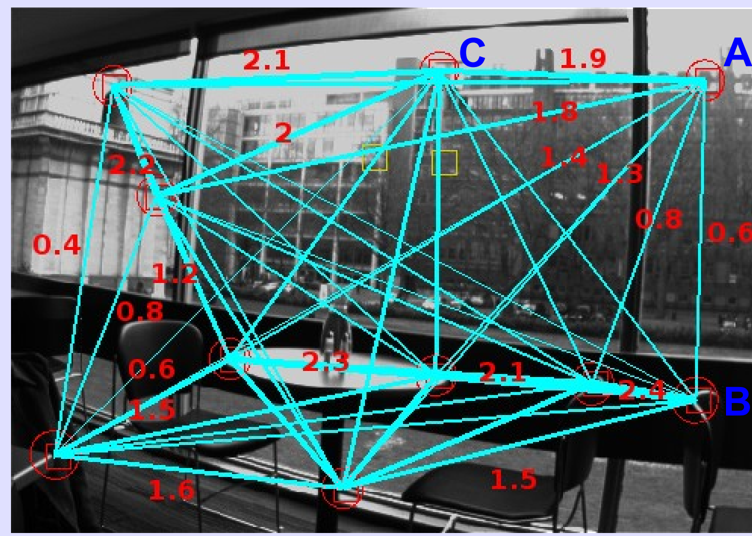
- Expected information gain of $\mathbf{z}_i$ given the exact state of $\mathbf{z}_j$:

If candidate measurements $\mathbf{z}_T \sim Normal(\hat{\mathbf{z}_T}, \mathtt{S})$ ⇨ $I(\mathbf{z}_i;\mathbf{z}_j) = \frac{1}{2}\log_2 \frac{|\mathtt{S}_{ii}||\mathtt{S}_{jj}|}{|\mathtt{S}_{ij}|}$

[Active Matching, ECCV 2008]:

Predicted MI of each feature used to guide sequential matching

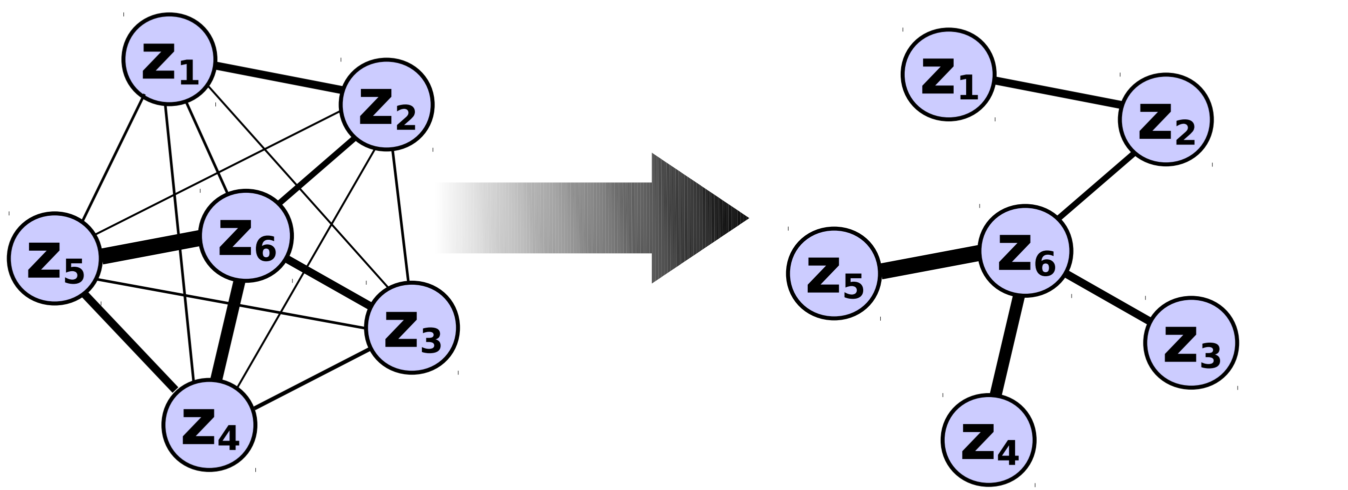
Here, use pairwise MIs to reveal the features' correlation structure



✓ Pairwise MI: absolute, normalised measure of degree of correlation, valid for any combination of feature types

Chow-Liu Tree

- Aim: unveil the hierarchy of correlations encoded in the complete MI graph ⇨ use a tree decomposition
- Chow, Liu 1968:
  The optimum approximation of a joint probability density with a 1st order dependency tree is the ***max. spanning tree*** of the MI graph:

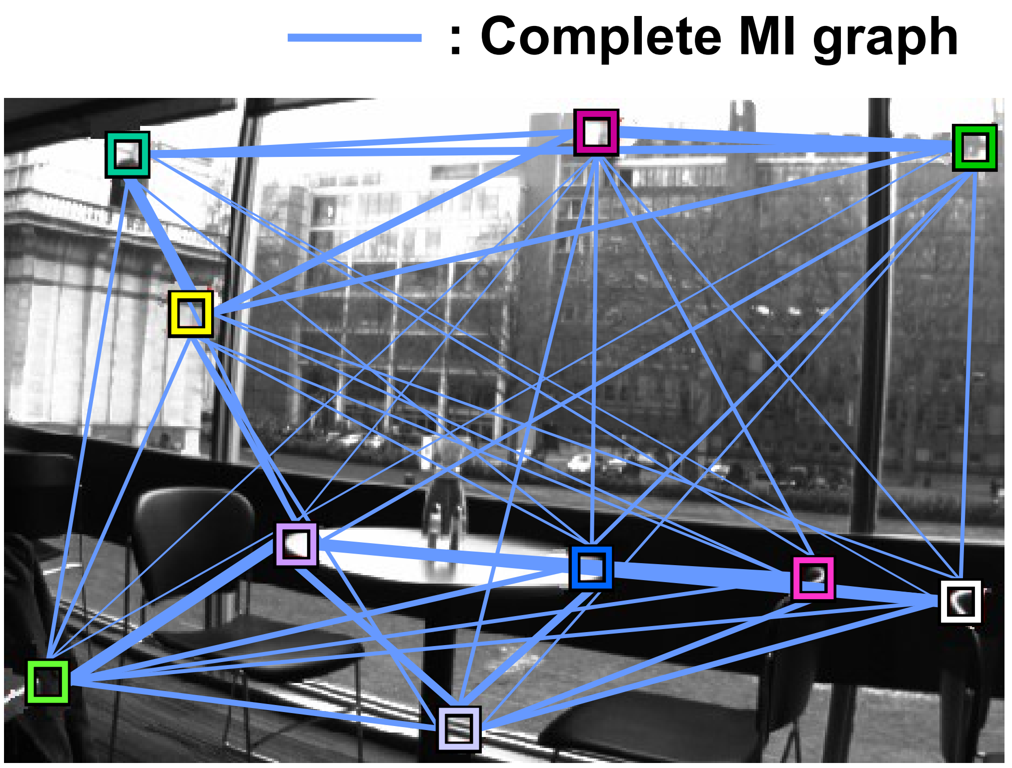


Complete MI graph

Chow-Liu Tree approximation

Imperial College London

Inferring Hierarchical Scene Structure

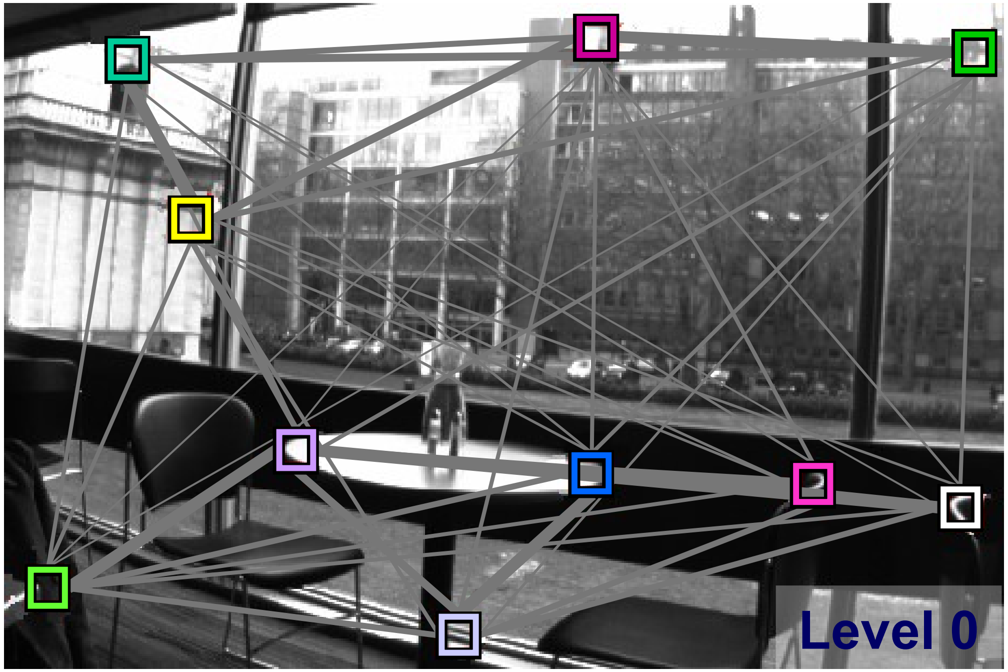


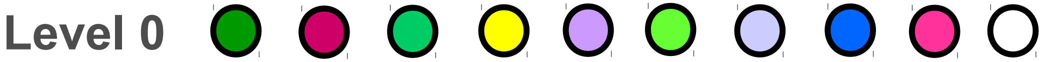

```
Hierarchy level h=0: each
feature in different submap

while(number of submaps > 1)

{

  • Grow each submap
    following the strongest,
    outgoing MI link

  • New level h = h+1,
    corresponds to new
    submaps
}
```

Inferring Hierarchical Scene Structure

Hierarchy level h=0: each feature in different submap

```
while(number of submaps > 1)
{
  • Grow each submap following the strongest, outgoing MI link
  • New level h = h+1, corresponds to new submaps
}
```





Inferring Hierarchical Scene Structure

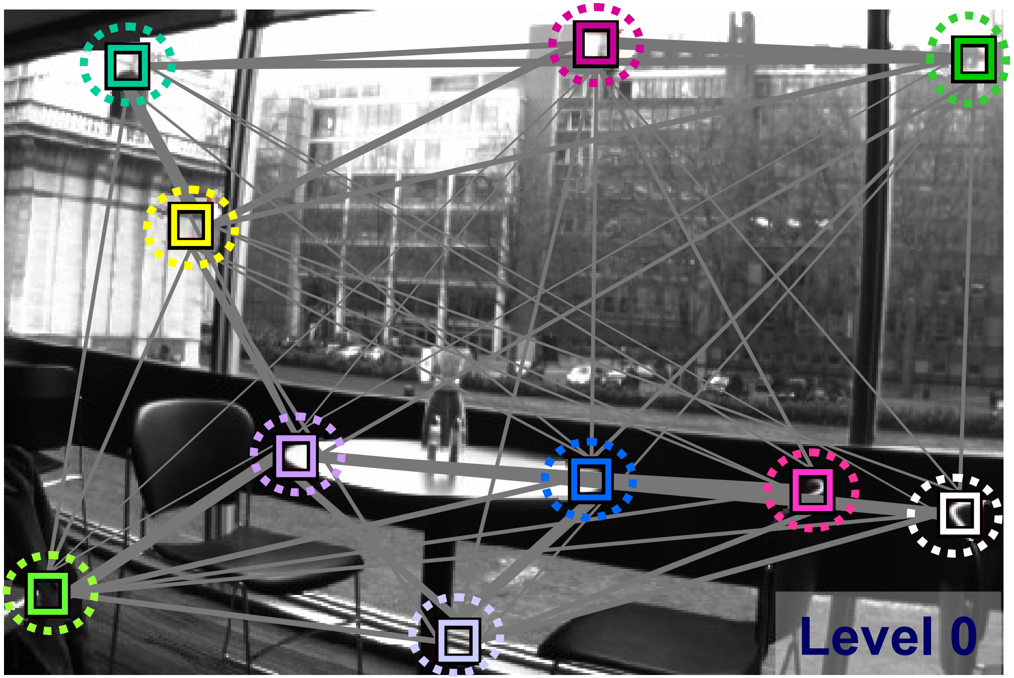
Hierarchy level h=0: each feature in different submap

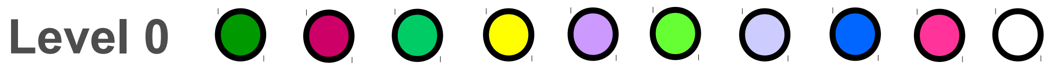
while(number of submaps > 1)

{

- Grow each submap following the strongest, outgoing MI link
- New level h = h+1, corresponds to new submaps

}





Imperial College London

Inferring Hierarchical Scene Structure

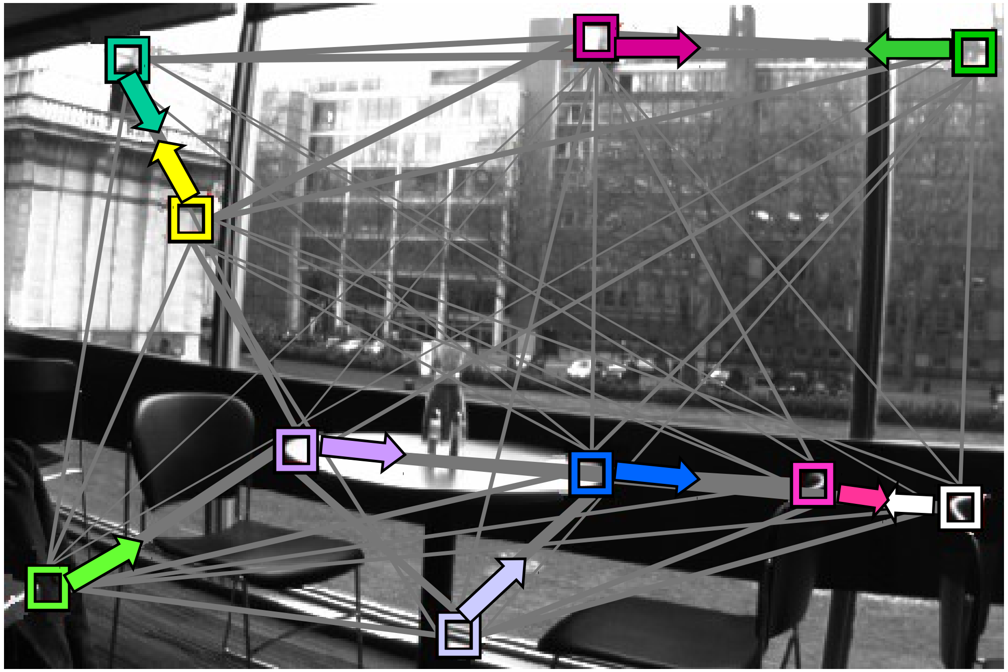
Hierarchy level h=0: each feature in different submap

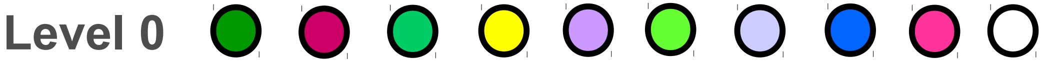
while(number of submaps > 1)

{

- **Grow each submap following the strongest, outgoing MI link**
- New level h = h+1, corresponds to new submaps

}





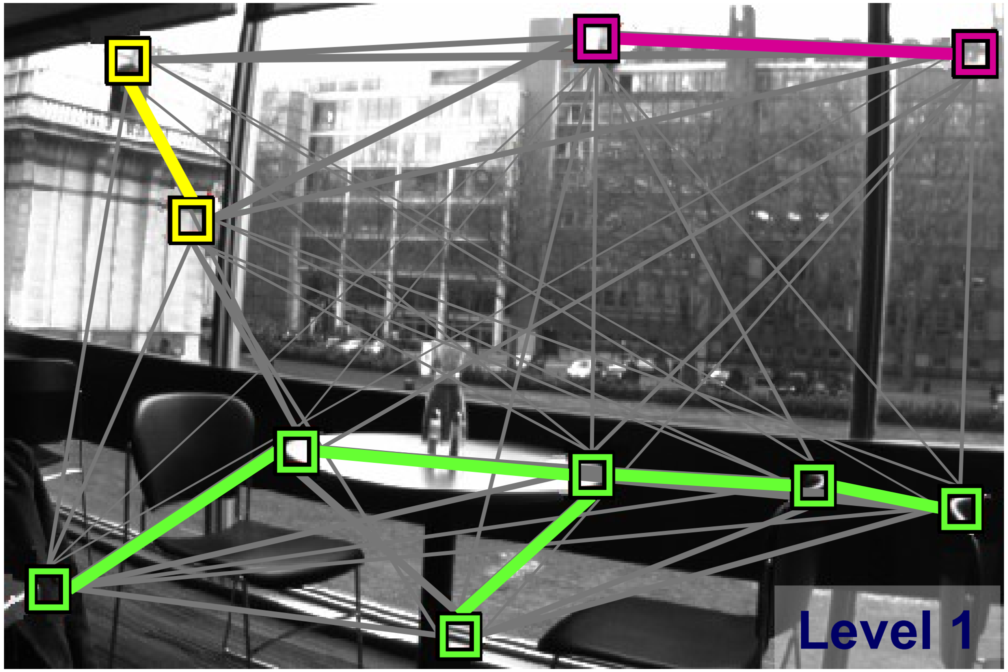
Inferring Hierarchical Scene Structure

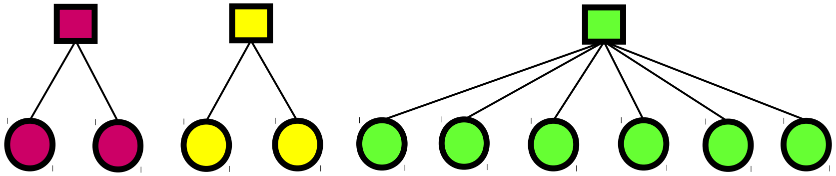
Hierarchy level h=0: each feature in different submap

```
while(number of submaps > 1)
{
```

- Grow each submap following the strongest, outgoing MI link
- **New level h = h+1, corresponds to new submaps**

```
}
```





Inferring Hierarchical Scene Structure

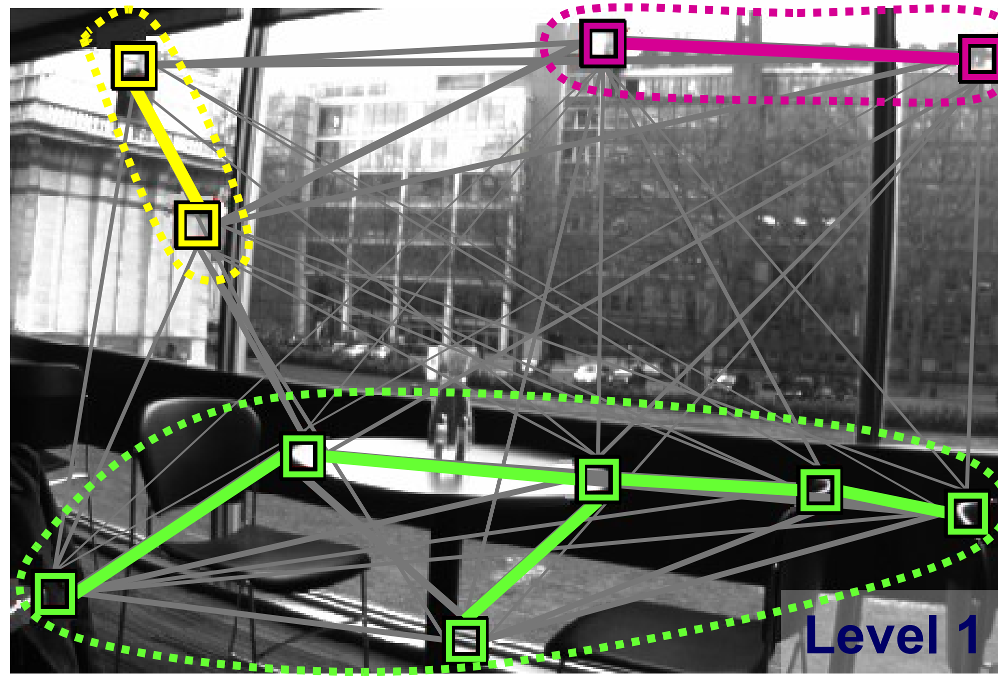
Hierarchy level h=0: each feature in different submap

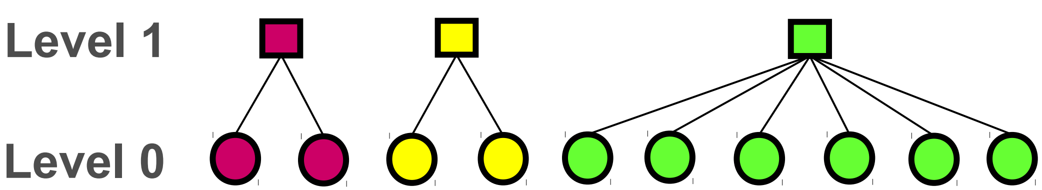
while(number of submaps > 1)

{

- Grow each submap following the strongest, outgoing MI link
- New level h = h+1, corresponds to new submaps

}





Inferring Hierarchical Scene Structure

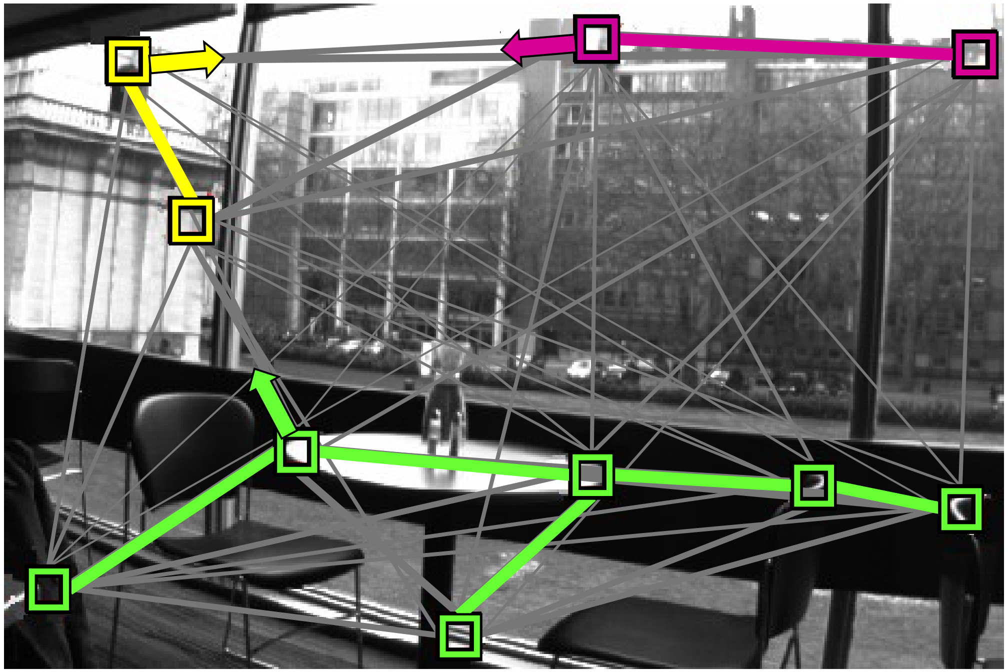
Hierarchy level h=0: each feature in different submap

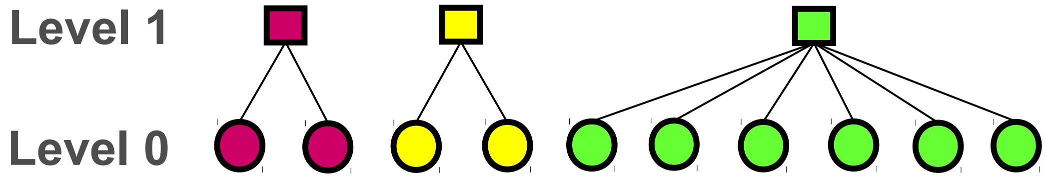
while(number of submaps > 1)

{

- **Grow each submap following the strongest, outgoing MI link**
- New level h = h+1, corresponds to new submaps

}

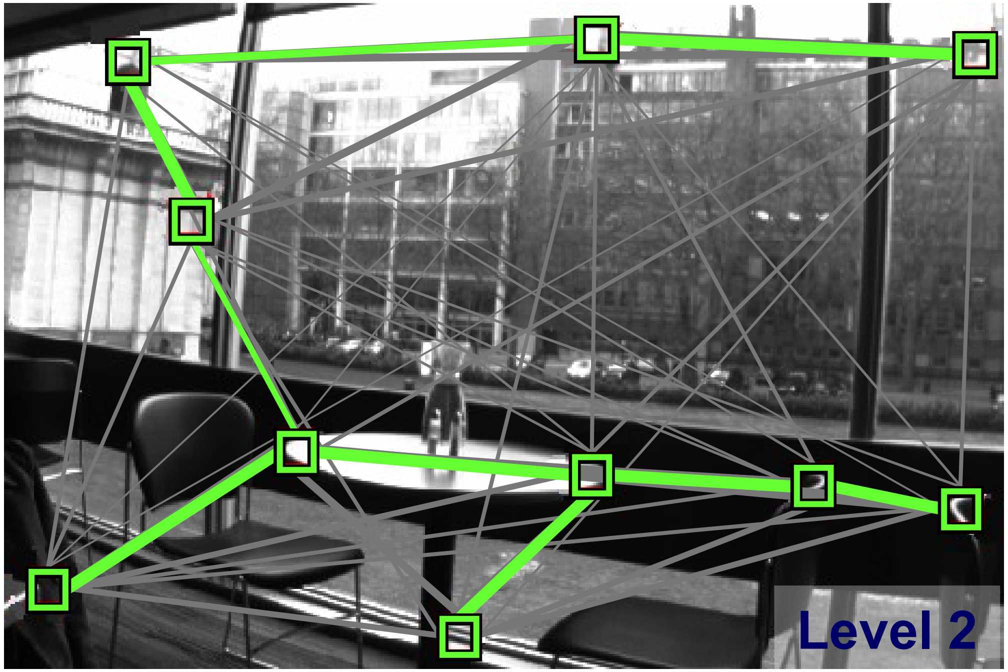




Imperial College London

Inferring Hierarchical Scene Structure

: Chow-Liu Tree

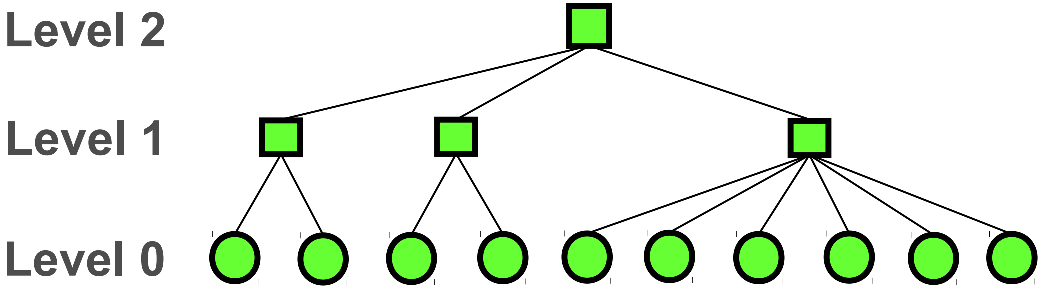


Hierarchy level h=0: each feature in different submap

```
while(number of submaps > 1)
{
```

- Grow each submap following the strongest, outgoing MI link
- **New level h = h+1, corresponds to new submaps**

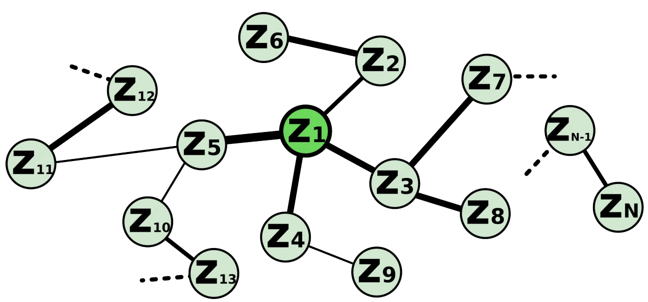
```
}
```

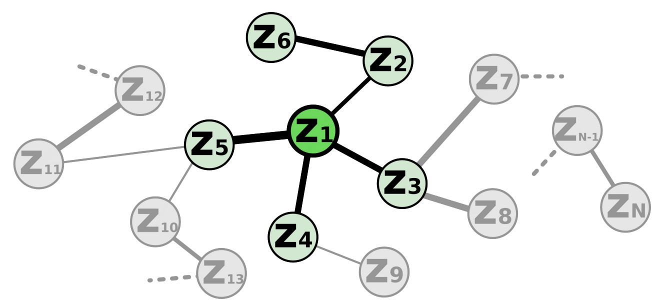


Imperial College London

CLAM: Chow-Liu Active Matching

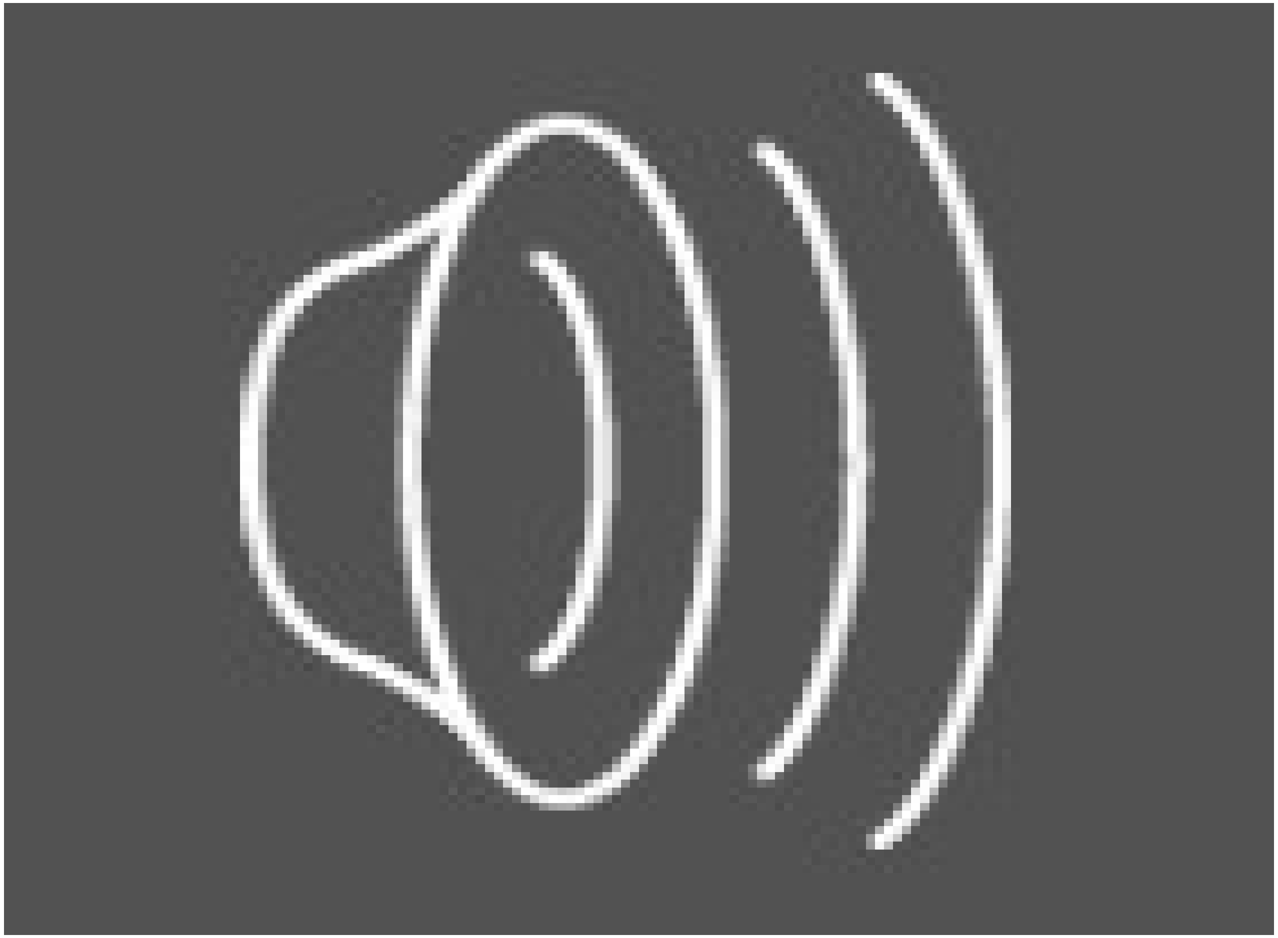
- Probabilistic approach to approximate AM
- **CL-tree** captures the most representative correlation structure
  ⇨ sparsify the joint distribution of features into the CL-tree
- Tree exhibits attractive properties:
  - ✓ BP to propagate update-messages in a single pass
  - ✓ Matching progressively partitioned into sub-trees
  - ✓ $\mathbf{MI}(\mathbf{z}_1;\mathbf{z}_2,\mathbf{z}_3,\ldots,\mathbf{z}_N) = \mathbf{MI}(\mathbf{z}_1;ne(\mathbf{z}_1)) = \mathbf{MI}(\mathbf{z}_1;\mathbf{z}_2,\mathbf{z}_3,\ldots,\mathbf{z}_6)$



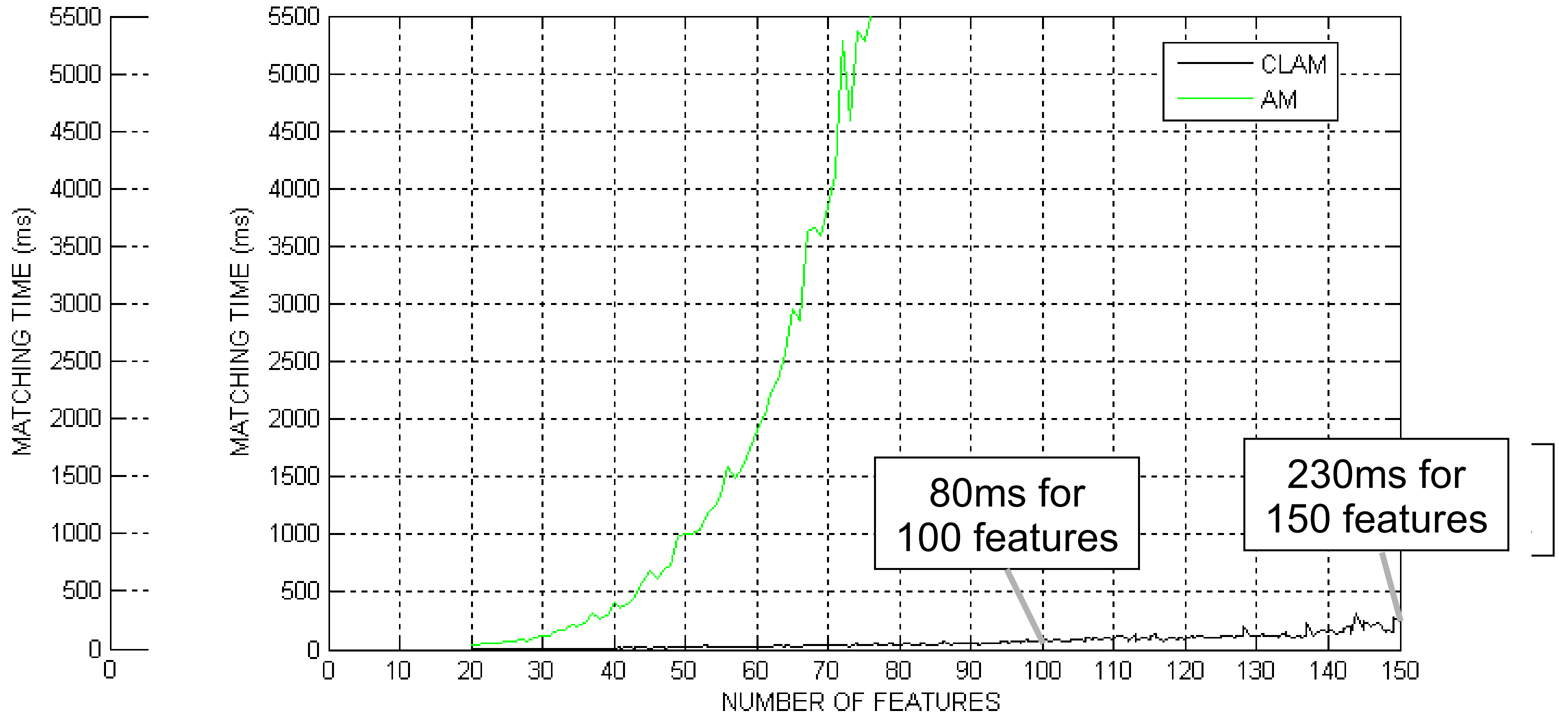


[Handa, Chli, Strasdat, Davison CVPR2010]

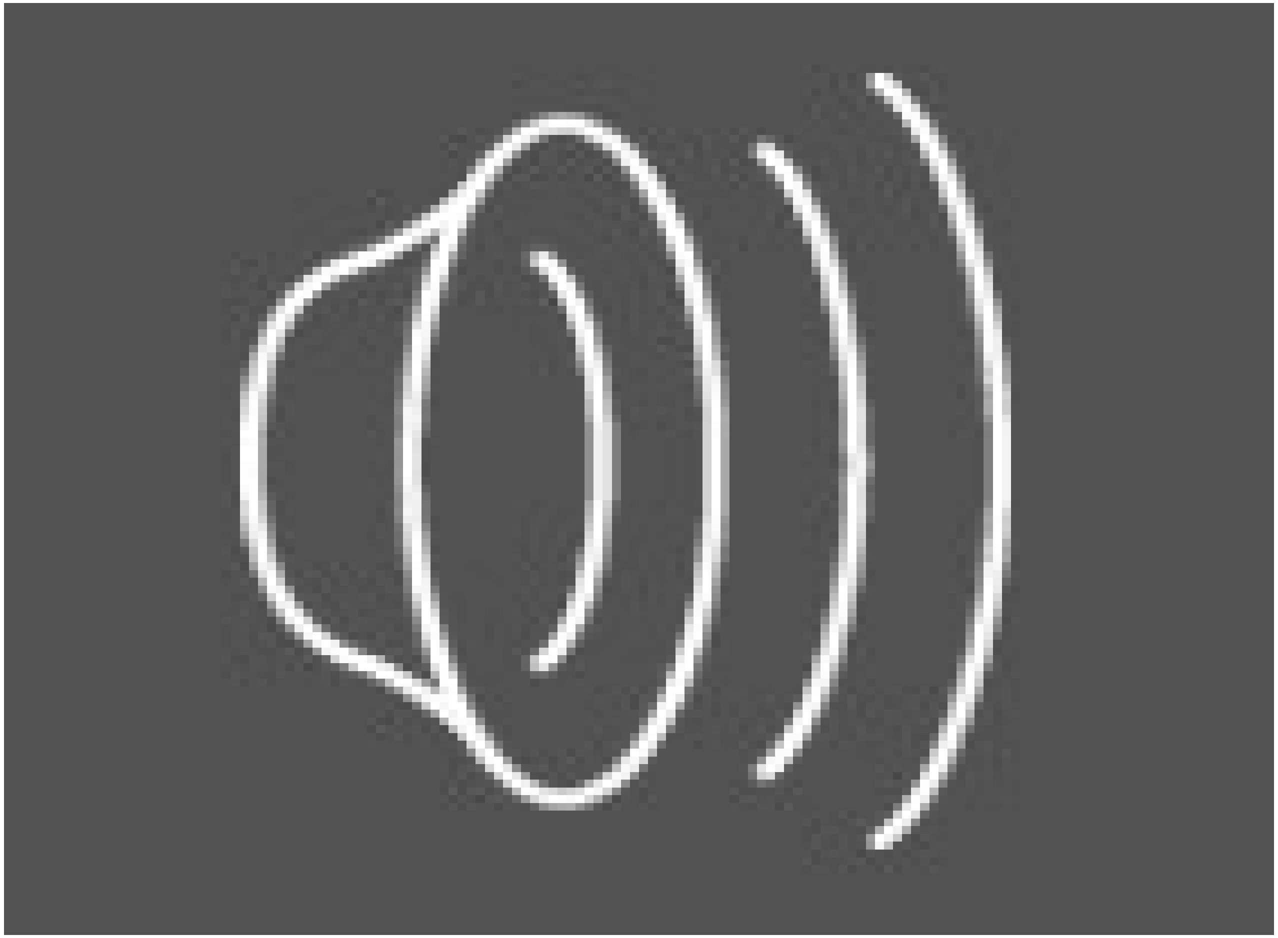
CLAM: an example



CLAM and SubAM: time performance



SubAM: Subset Active Matching



Imperial College London

Conclusion & Future directions

- ✓ **AM**: fully Bayesian **robust, multi-hypothesis** feature matching
- ✓ Takes full account of the input priors available in matching tasks.
- ✓ **CLAM** & **SubAM**: powerful algorithms for **dense, real-time matching**

What's next?

- These methods should adapt well to tasks such as matching at very high frame-rates where priors are very strong.
- Can we go even further to match every single pixel? – only possible with heavyweight ***optical flow*** techniques so far