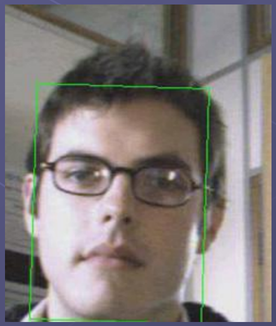
UNIVERSITY OF CAMBRIDGE

A Sparse Probabilistic Learning Algorithm for Real-Time Tracking

Oliver Williams
University of Cambridge

Andrew Blake
Microsoft Research, Cambridge

Roberto Cipolla
University of Cambridge

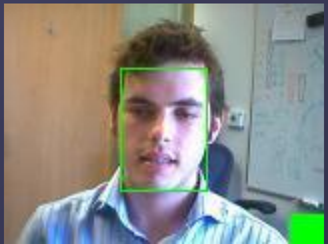


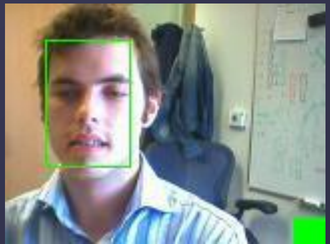
Robust Tracking

- Self starting
- Self recovering
- Efficient

object detector

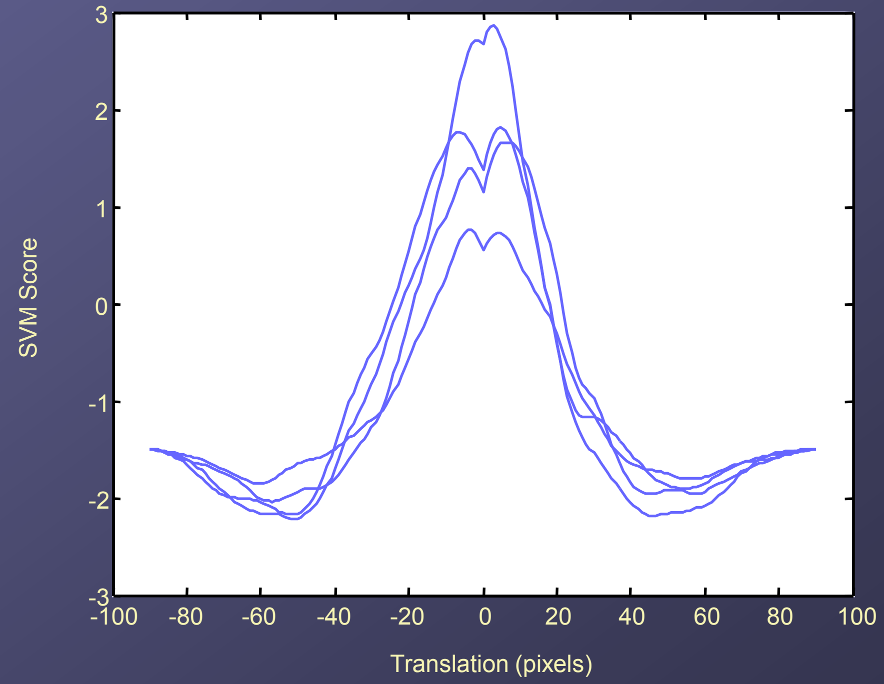
exploit temporal coherence





Background: Support Vector Tracking

- Avidan [CVPR 2001]
- SVM on object class
- Treat score as function of position



$$f_{svm}(\mathbf{x}_t) = f_{svm}(\mathbf{x}(\mathbf{u}_t))$$

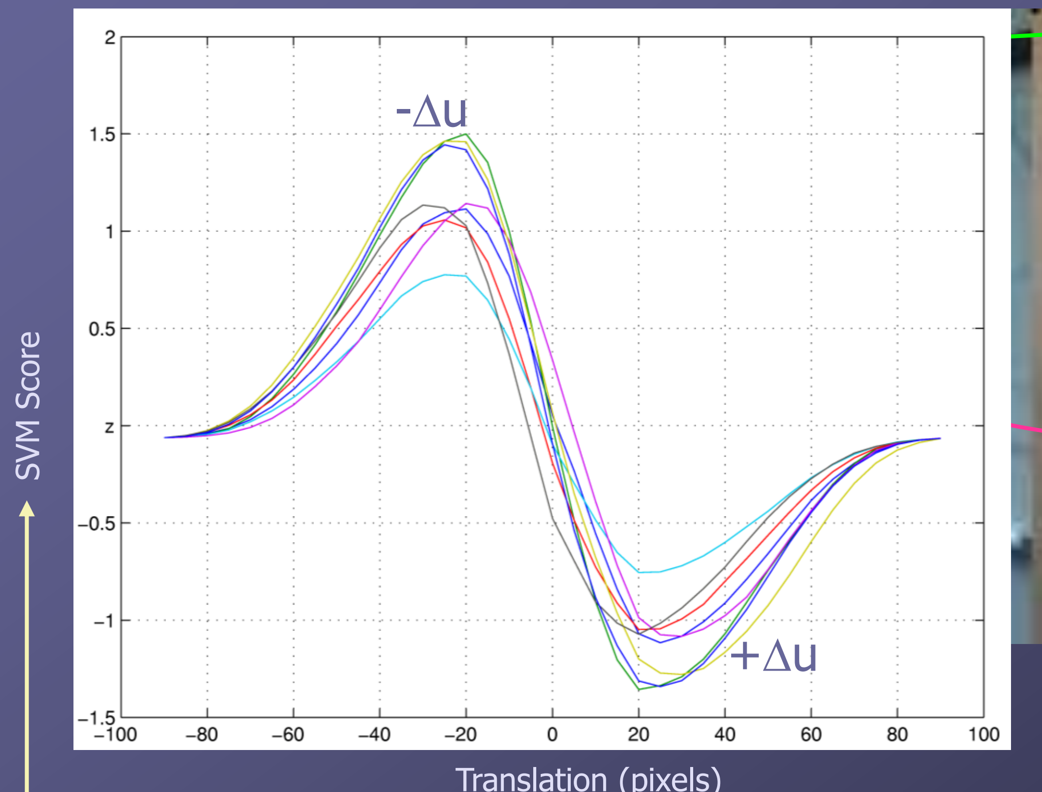
$$\approx f_{svm}(\mathbf{x}_{t-1} + \delta\mathbf{u} \cdot \nabla\mathbf{x}_t)$$

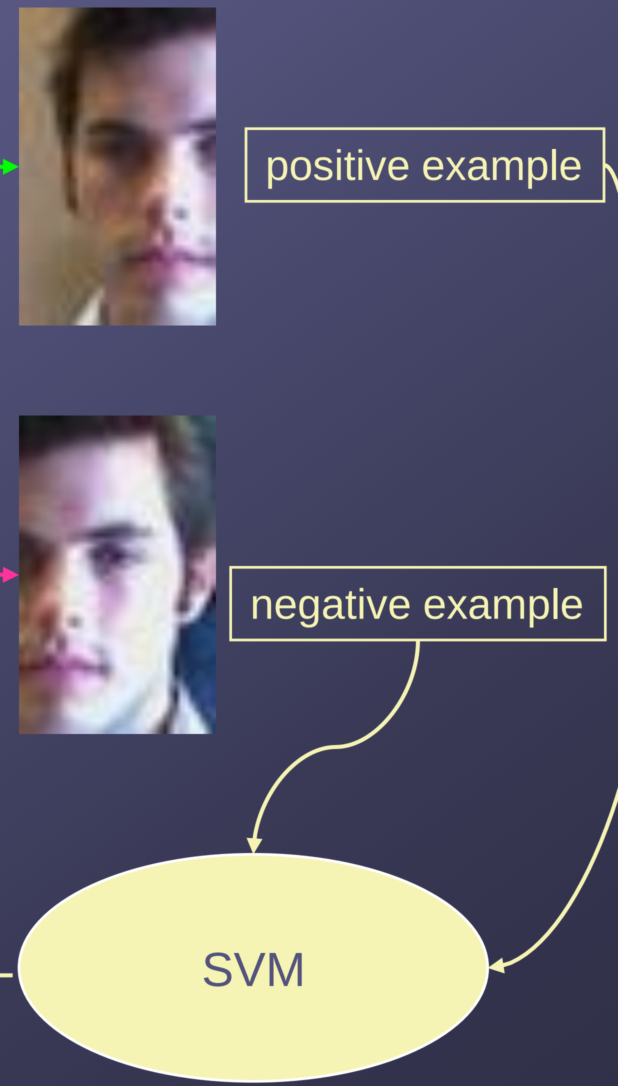
subimage

image gradient

Training for displacement

- Learn right versus left





- Calibrate state update from SVM score

Temporal Fusion

- Probabilistic
  - Statistical filter
- Want sparsity
  - Computational efficiency

Relevance Vector Machine

- Tipping [NIPS 2000]
- SVM in Bayesian setting
- Learns continuous function from training set $\{\mathbf{z}_i, t_i\}$

RVM Evaluation Equation

test subimage

weights

training subimage

$$y(x) = \sum_{i=1}^{N} w_i k(x, z_i) + w_0$$

kernel function

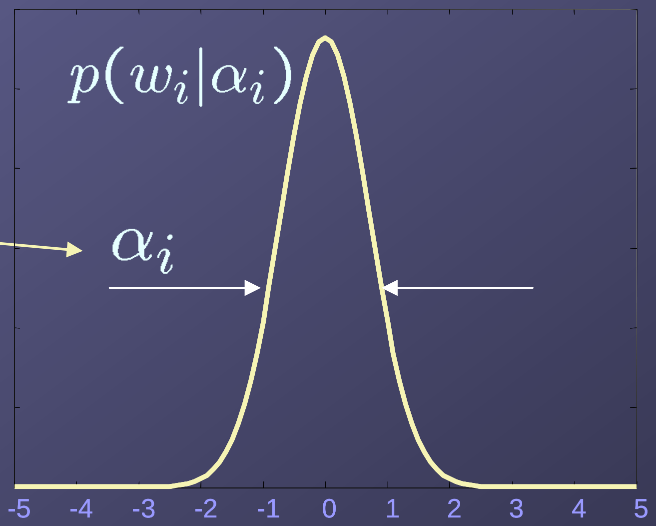
- Setting some w_i to zero → **sparse** solution

Bayesian Training

- Sparsity encouraged by zero-mean **prior** :

$$w_i \sim \mathcal{N}(0, \alpha_i)$$

hyperparameter



- Training data modelled by Gaussian **likelihood** function

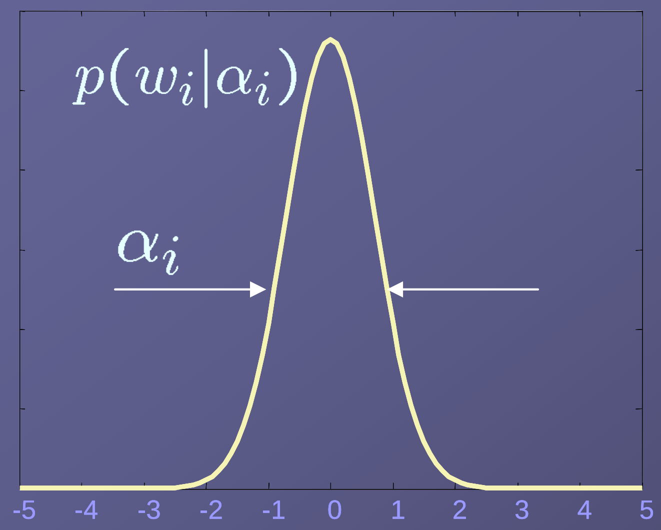
$$t_i = y(z_i) + \epsilon(\sigma^2)$$

provided target

"true" underlying value

noise parameter

Occam's Razor



$$t_i = y(z_i) + \epsilon(\sigma^2)$$

- Sometimes, lower Occam penalty paid by removing an example…

$$\alpha_i \to 0$$

- …and explaining data with more noise

$$\sigma^2 \uparrow$$

Relevance Vector Machine

- Posterior

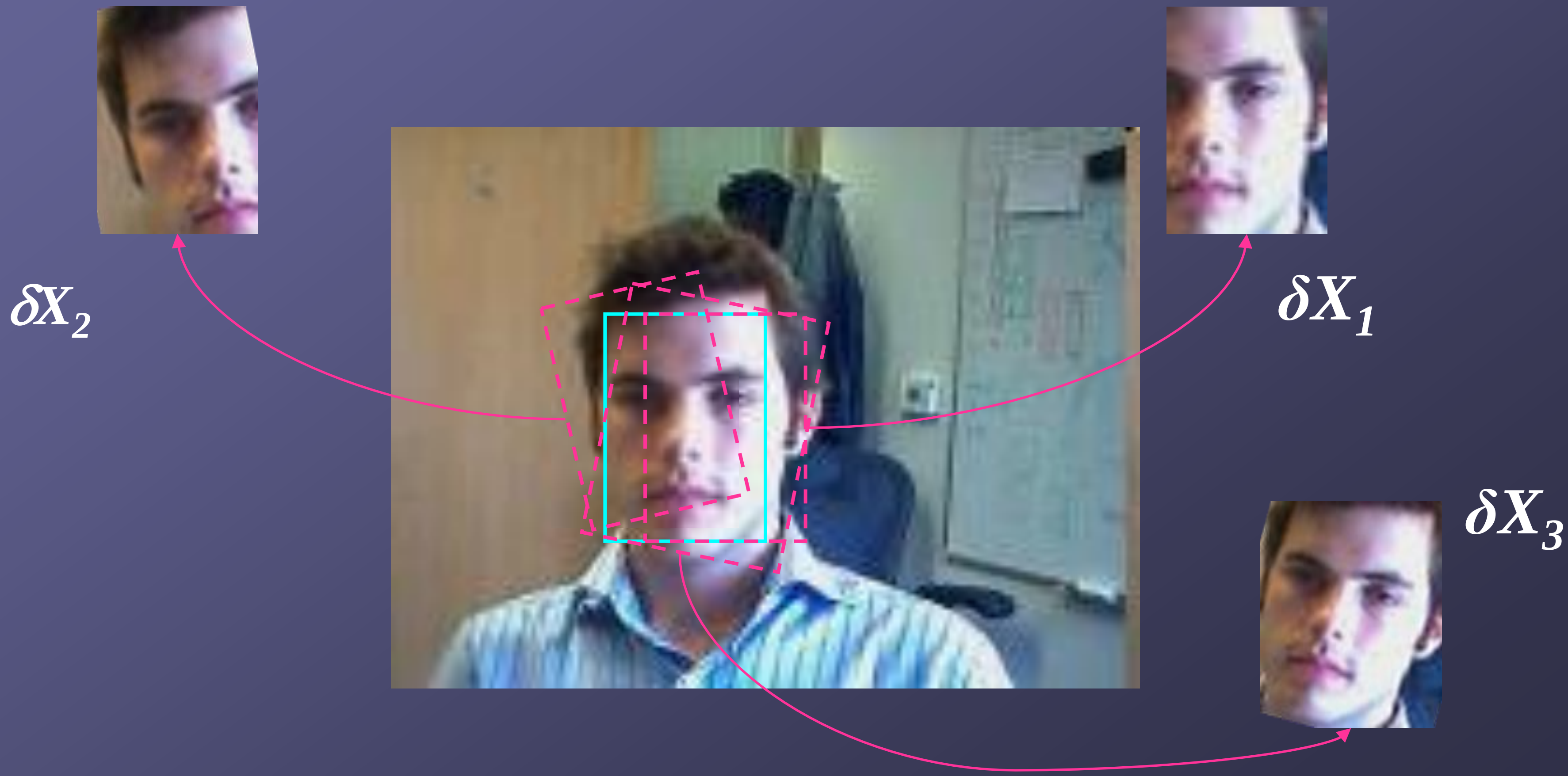
$$p(\mathbf{w}|\mathbf{t}, \{\mathbf{z}_i\}, \sigma^2, \{\alpha_i\})$$

- Train RVM by finding **hyperparameters**
- Maximise **marginal likelihood** $p(\mathbf{t}, \{\mathbf{z}_i\}|\sigma^2, \{\alpha_i\})$
- “Prune” vector i when $\alpha_i \to 0$

UNIVERSITY OF CAMBRIDGE

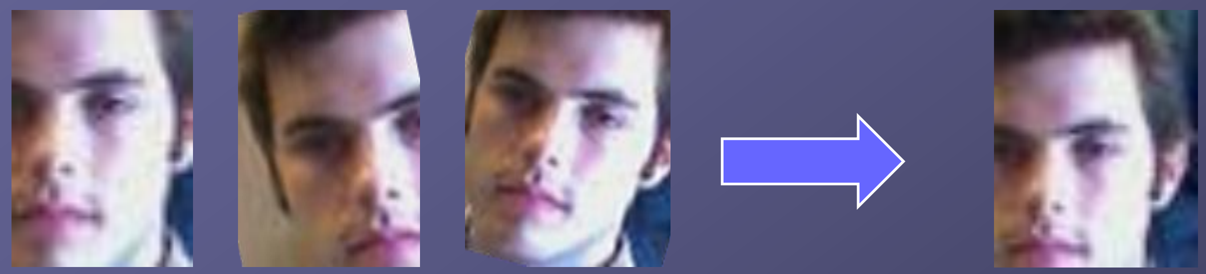
Creating a Training Set

- Select a few "seed" stills
- Simulate translation, scaling and rotation
  - → labelled training set



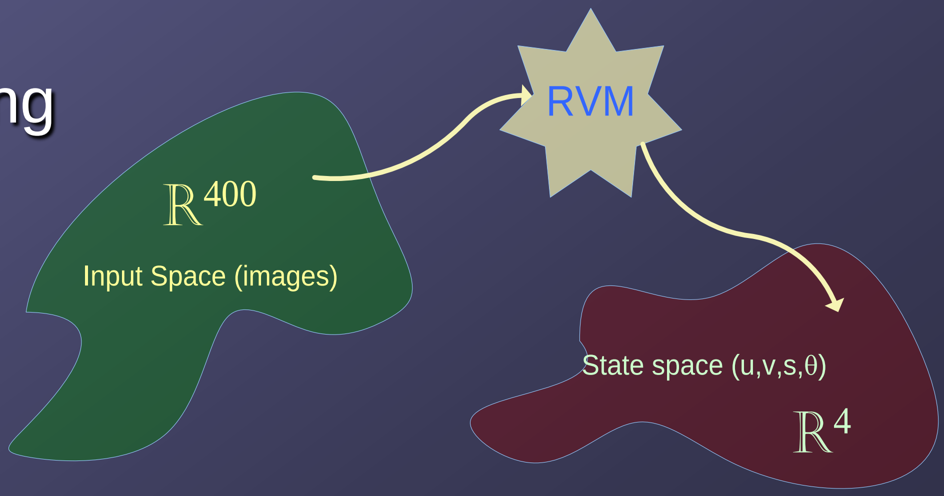
RVM Tracking

$\{z_i\}$



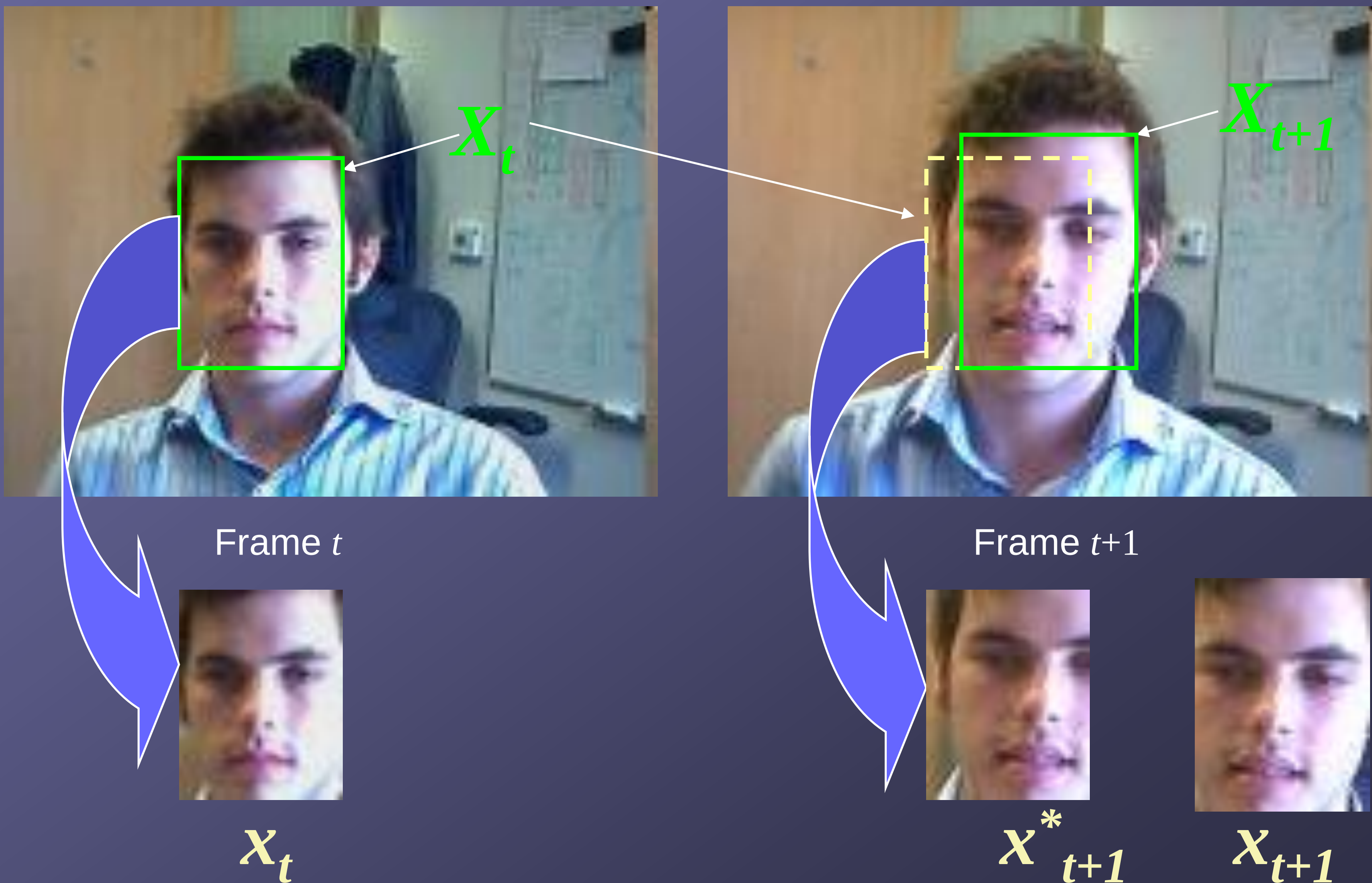
$\{t_i\}$ δX_1 δX_2 δX_3 δX_N

4D RVM learns mapping



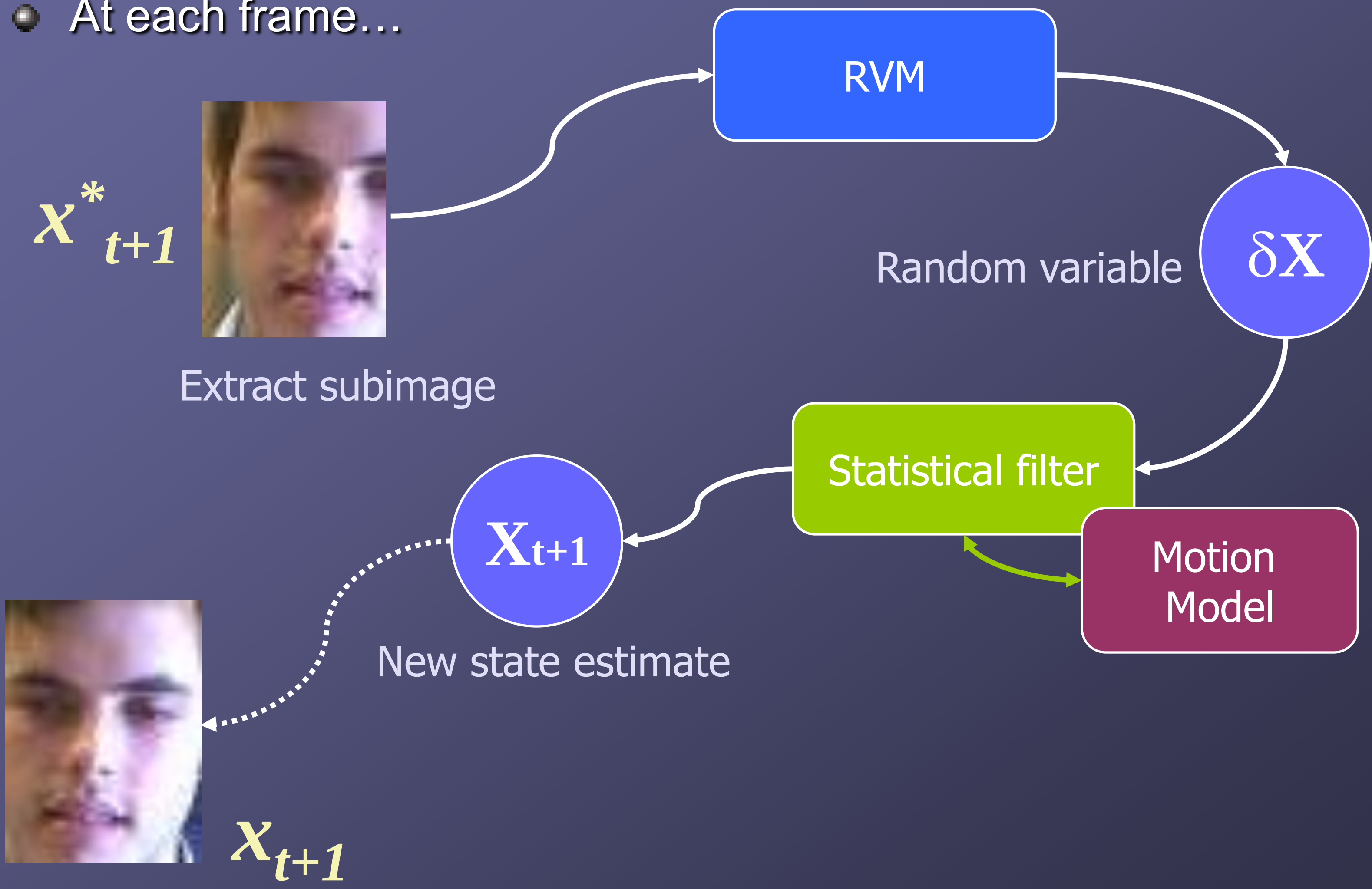
UNIVERSITY OF CAMBRIDGE

RVM Tracking



RVM Tracking

- At each frame…



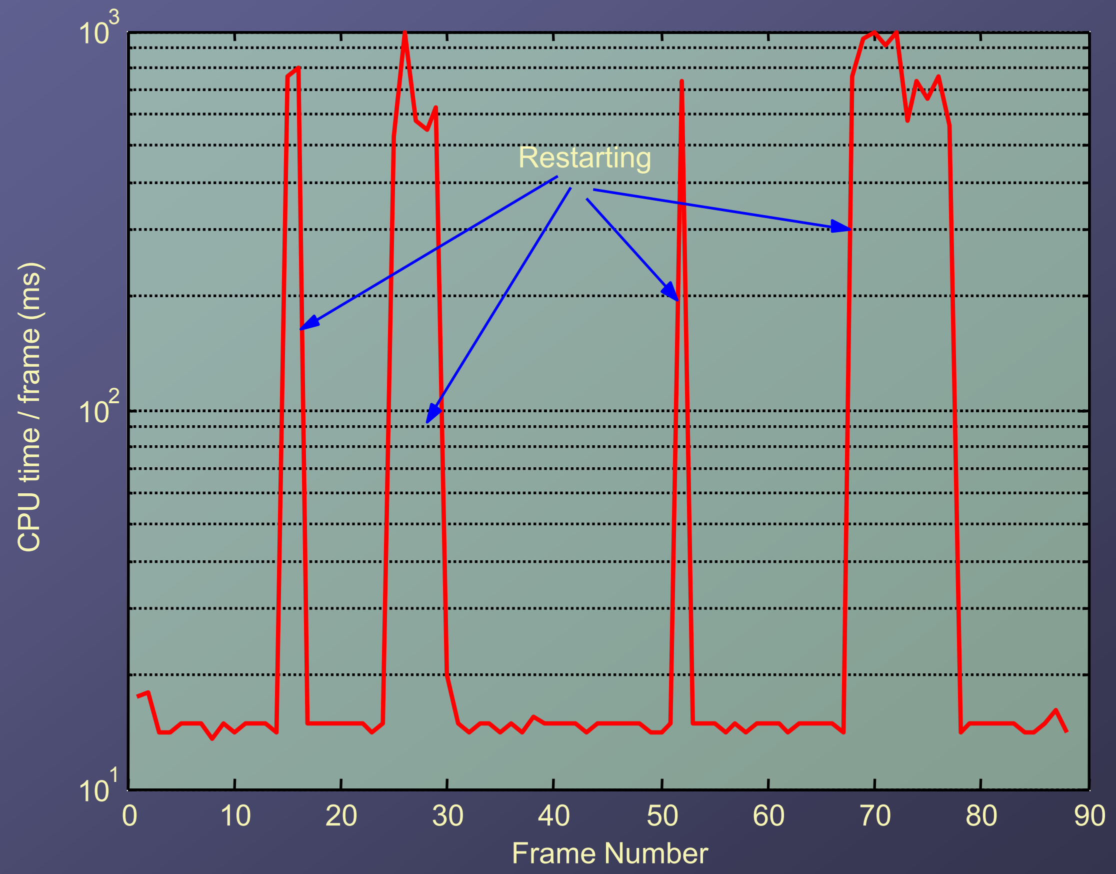
Initialization & Recovery

- Algorithm trains from “seed” subimages
  - Provided by localisation algorithm

- Localisation algorithm also used for
  - Initialization (frame 0)
  - Validation
  - Re-initialization

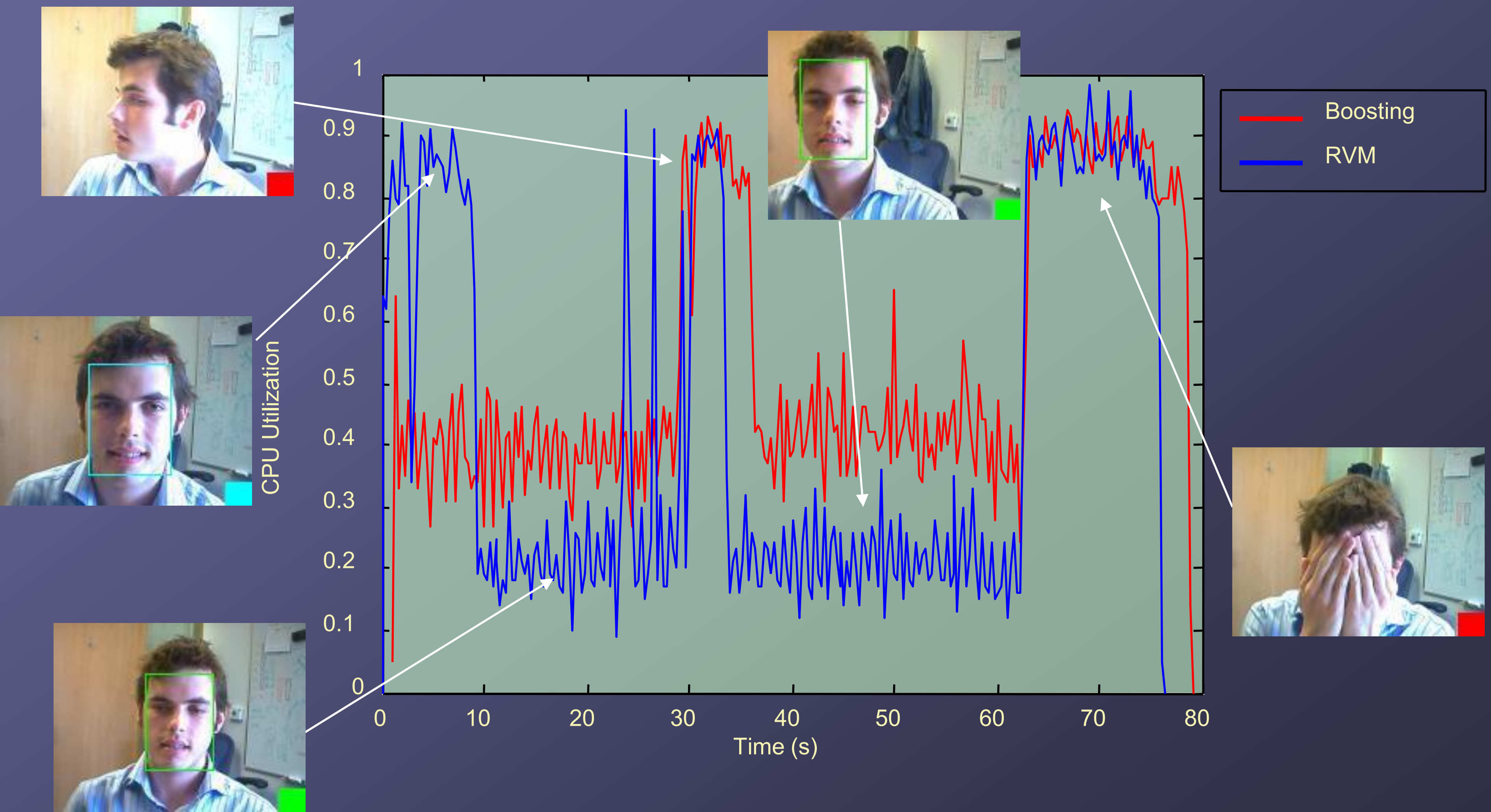
Results: Computational Efficiency

- Why track?



Results: Computational Efficiency

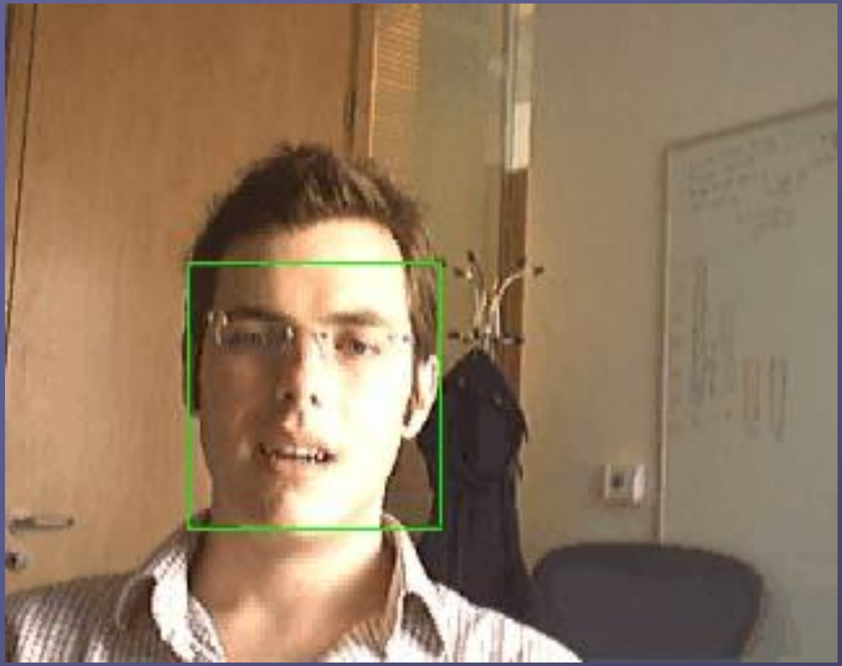
- Localize with boosting [Viola, Jones 2001, Li et al. 2002]



UNIVERSITY OF CAMBRIDGE

Automatic Camera Management

- Use position/scale information to control digital **pan** and **zoom**



Conclusions

- Future Work
  - RVM sparsity-smoothness tradeoff
  - Robustness to illumination and occlusion
  - Tracking 3D motion

- Hybrid RVM tracking is…
  - Self starting SVM/Boosting detector
  - Self recovering SVM/Boosting validation
  - Efficient Temporal fusion + sparsity

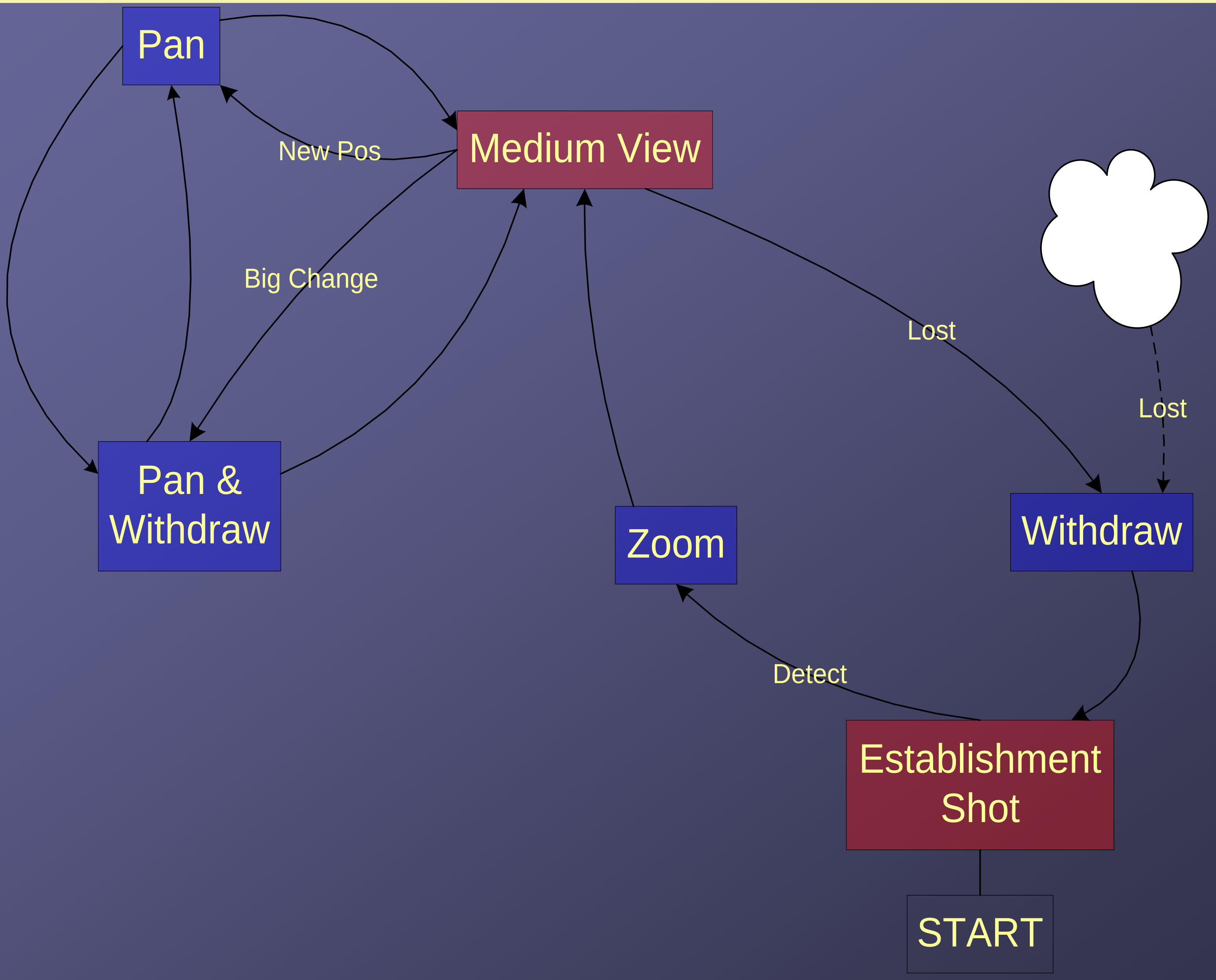
Questions?

UNIVERSITY OF CAMBRIDGE

Results: Cars

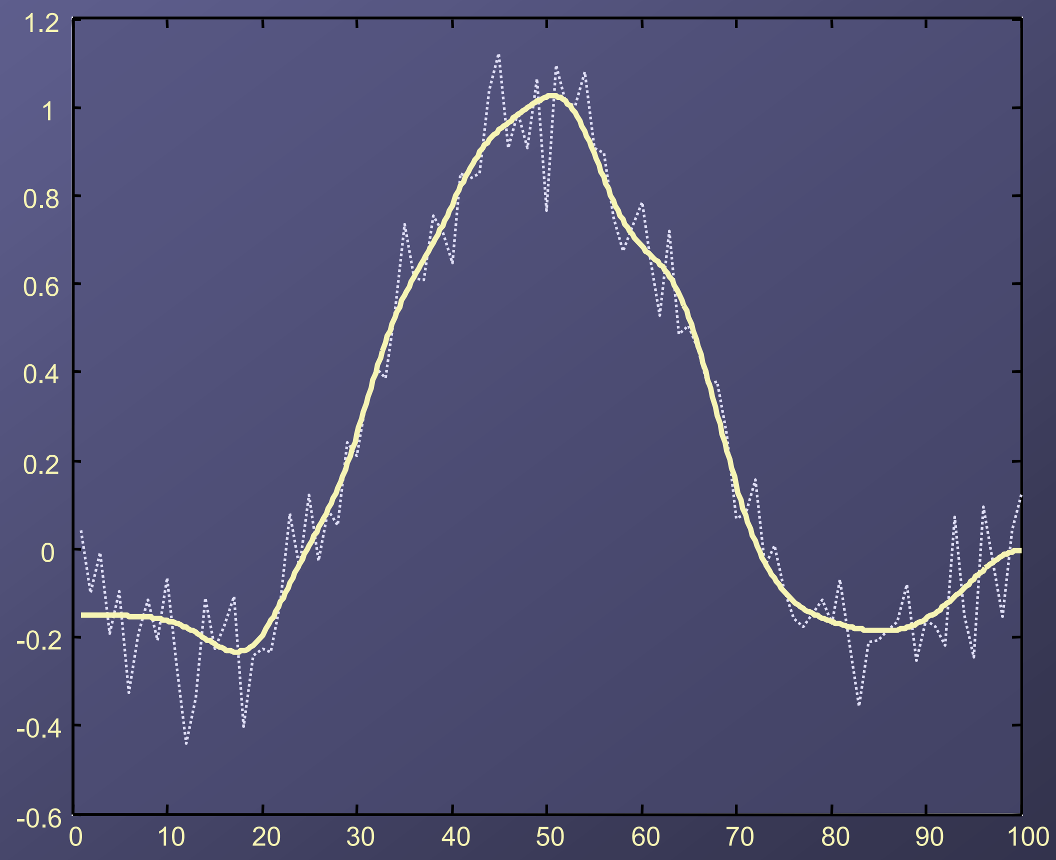
- Algorithm is not specific to any class of objects

cinematographer

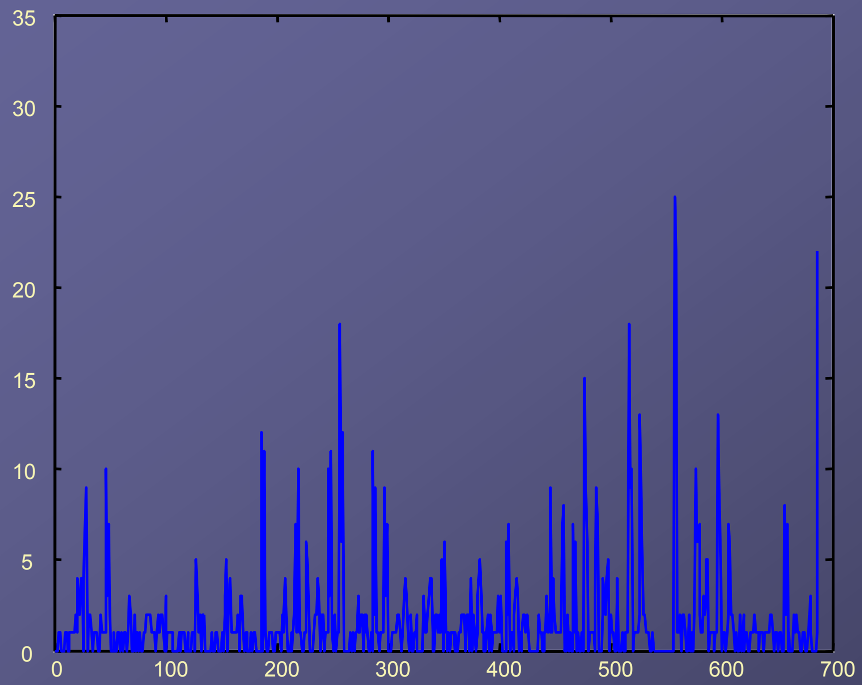


1D example

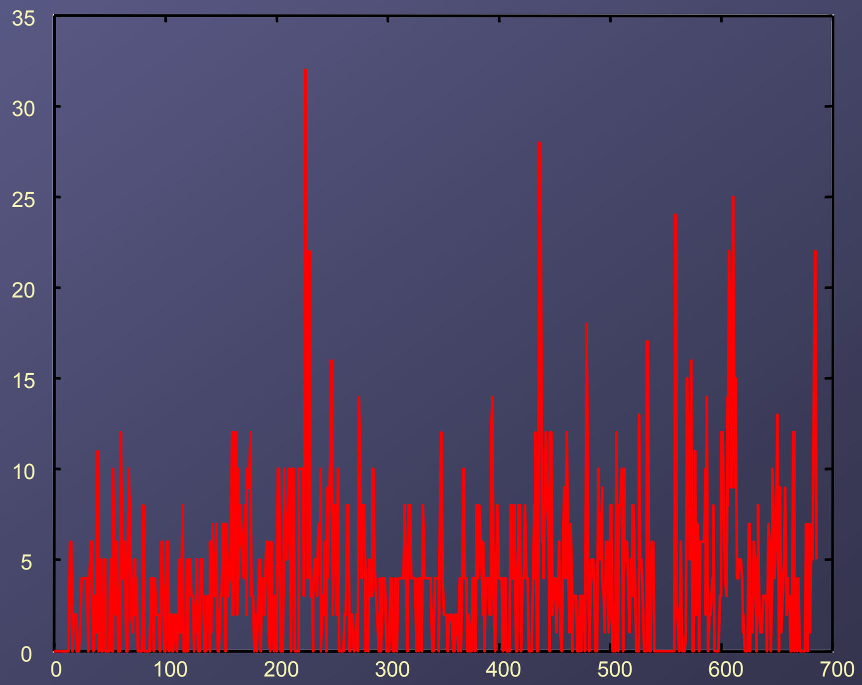
- Tutorial: toy data
- 100 Training examples
- 12 Relevant vectors remain



Results: Smoothness



RVM



Boosting