

Robust Background Estimation for Complex Video Sequences

Dirk Farin,
Peter H. N. de With,
Wolfgang Effelsberg

University of Mannheim
Dept. Computer Science IV
L 15,16, 68131 Mannheim, Germany
farin@uni-mannheim.de

The Background Estimation Problem

- The video-object segmentation problem gets much easier if a pure background image is known.
- Often, it is not possible to record an explicit background image.
 - background changes slowly (sunlight changes direction)
 - foreground objects cannot be removed (hall with many people)
- In the following, we assume a non-moving camera. But the algorithm can still be applied by prepending a camera motion compensation step.

Previous Work: Continuous Update Algorithms

- Idea

- a current background image is maintained;
- background image b_t is updated by (weighted) averaging with new input images f_t

$$b_{t+1} = (1 - \alpha) \cdot b_t + \alpha f_{t+1}$$

- Often, α is adjusted to motion strength.
- Objects appear as shadows unless $t_{bkg} \gg t_{fgr}$.

- Motion adaptive update, e.g.:

- Andrea Cavallaro et al., „Video object extraction based on adaptive background and statistical change detection“, VCIP 2000

- Kalman filter formulation

- Christof Ridder et al., „Adaptive background estimation and foreground detection using Kalman-filtering“, ICRAM 1995



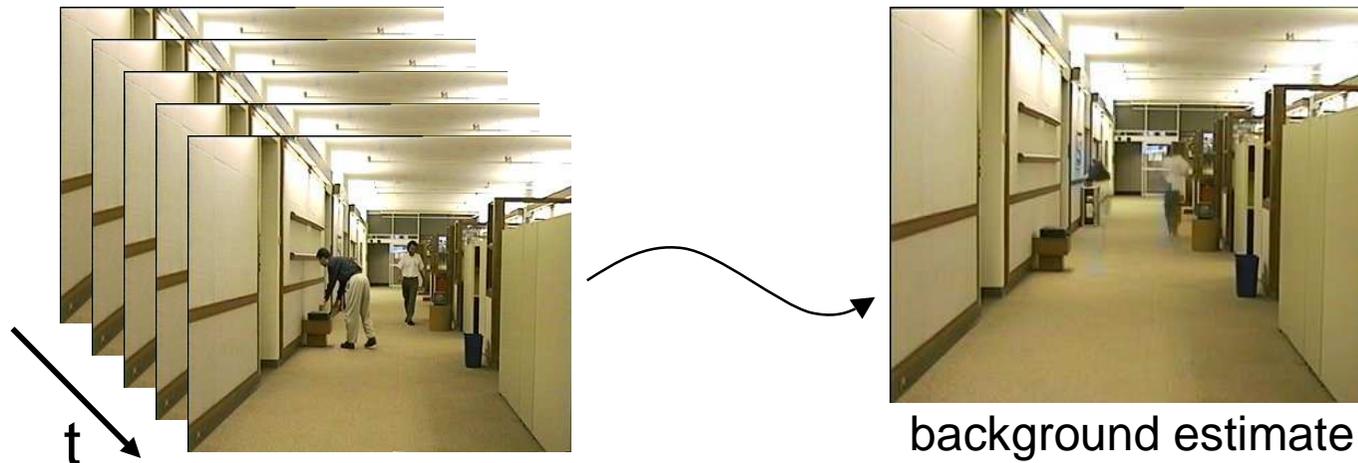
demo: constant α



demo: motion-adaptive α

Previous Work: Temporal Median Filtering

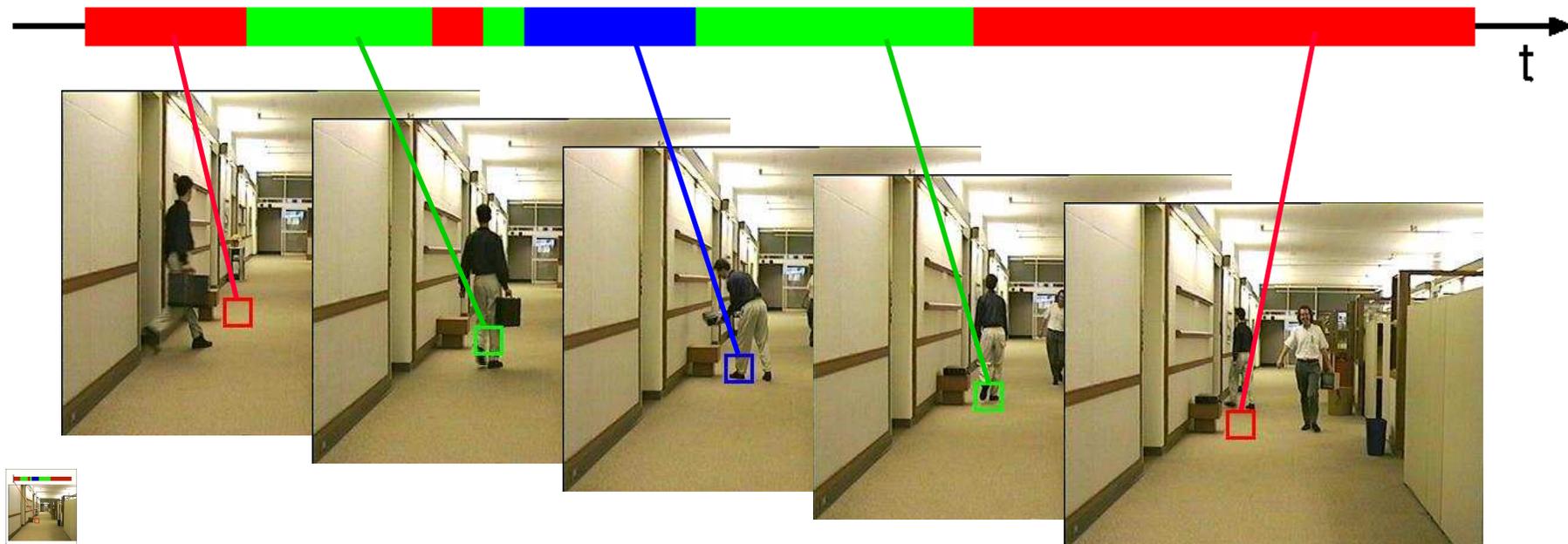
- Algorithm
 - pixel-wise median filter is applied in temporal direction
 - filter output is background image estimate
- Less blurring, but still requires that $t_{bkg} > t_{fgr}$.



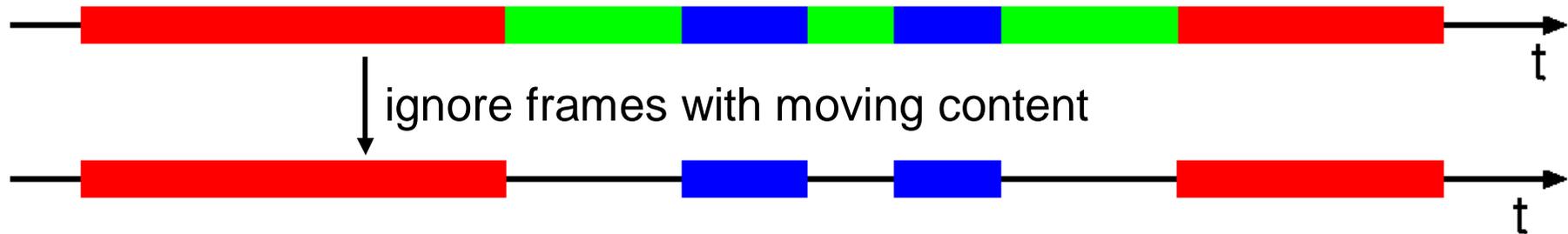
- M. Massey and W. Bender: „Salient stills: Process and practice“, IBM Systems Journal, 1996

A Closer Look at the Data

- Observe a small image block over some time. We can distinguish three classes of content:
 - static background
 - moving foreground
 - foreground which appears static within a small time range



Moving Content Exclusion



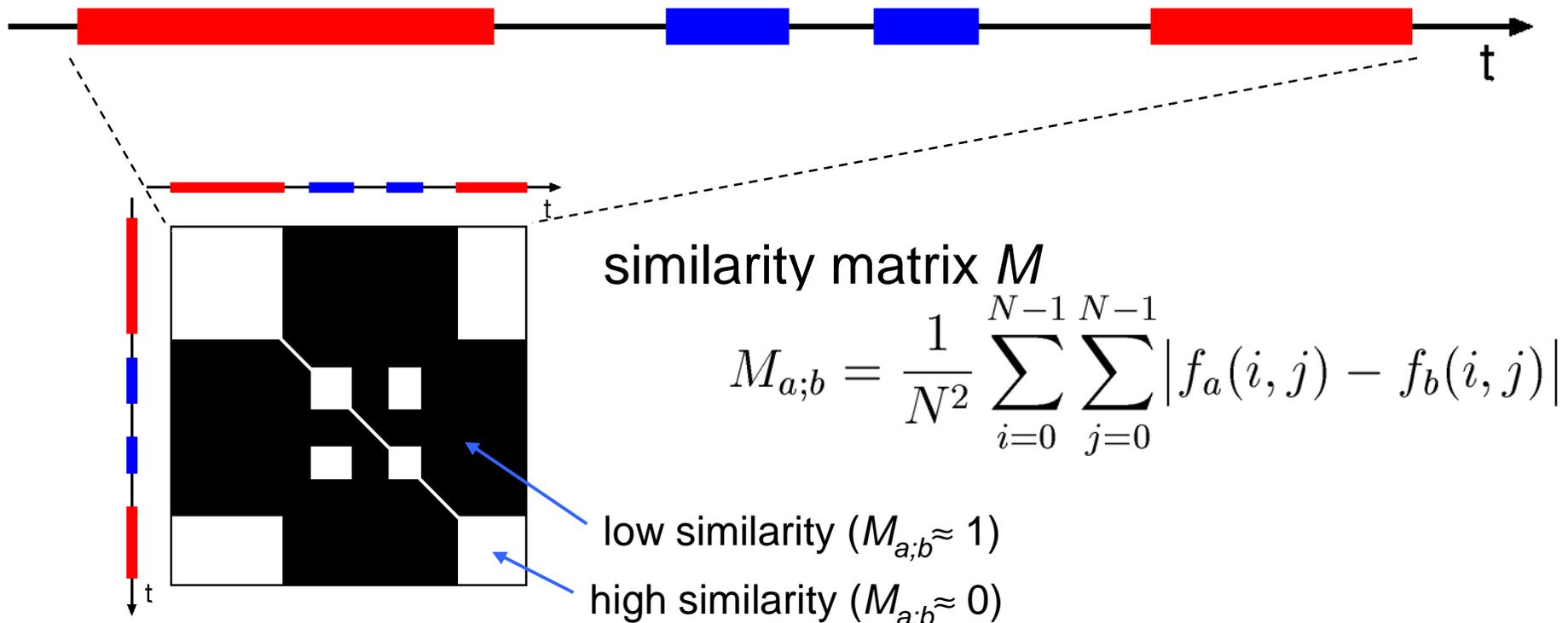
- Moving areas cannot be part of the background.
- Detect motion with simple block matching algorithm.
- We assume that block content is non-moving if

$$SAD(0, 0) < 90\% \cdot SAD(dx, dy) \quad dx \text{ or } dy \neq 0$$



Similarity Matrix

- We still have to differentiate between **static background** and **quasi-static foreground**.

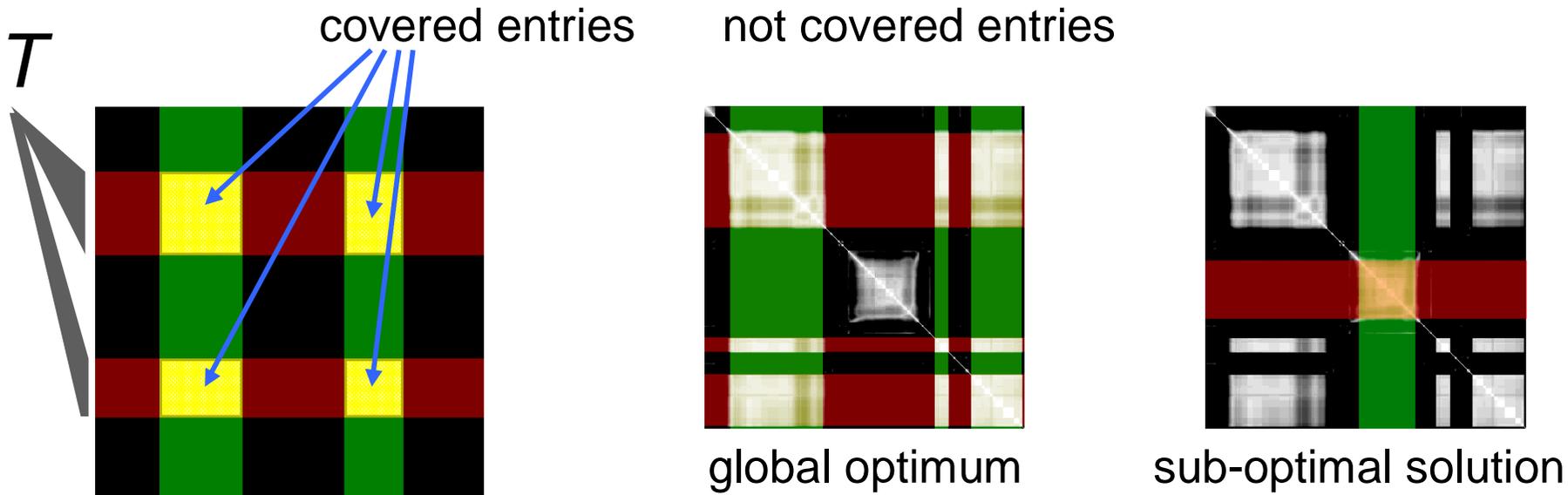


- Each matrix element $M_{a;b}$ contains similarity of two blocks at the same spatial position, but at different times a and b .

Matrix-Decomposition

- Find largest subset of frames T that contains static content.
 - Matrix entries covered by T should show low differences.
 - Matrix entries not covered should show high differences.

$$\min_T \underbrace{\sum_{a,b \in T} M_{a;b}}_{\text{covered entries}} + \underbrace{\sum_{a \notin T \vee b \notin T} (1 - M_{a;b})}_{\text{not covered entries}}$$

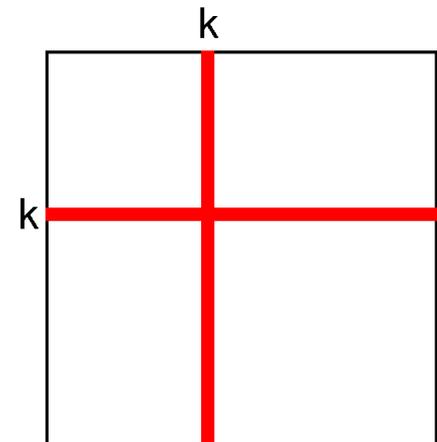


Decomposition Algorithm (1/2)

$$C = \sum_{a,b \in T} M_{a;b} + \sum_{a \notin T \vee b \notin T} (1 - M_{a;b})$$

- Change of cost if $k \notin T$ is added to T :

$$\begin{aligned} \Delta C_{+k} &= 2 \left(\underbrace{\sum_{a \in T} M_{a,k} + \sum_{a \notin T} (1 - M_{a,k})}_{\text{new costs}} - \underbrace{\sum_{a \in \{1, \dots, L\}} (1 - M_{a;k})}_{\text{old costs}} \right) \\ &= 2 \left(\sum_{a \in T} (2M_{a;k} - 1) \right) \end{aligned}$$



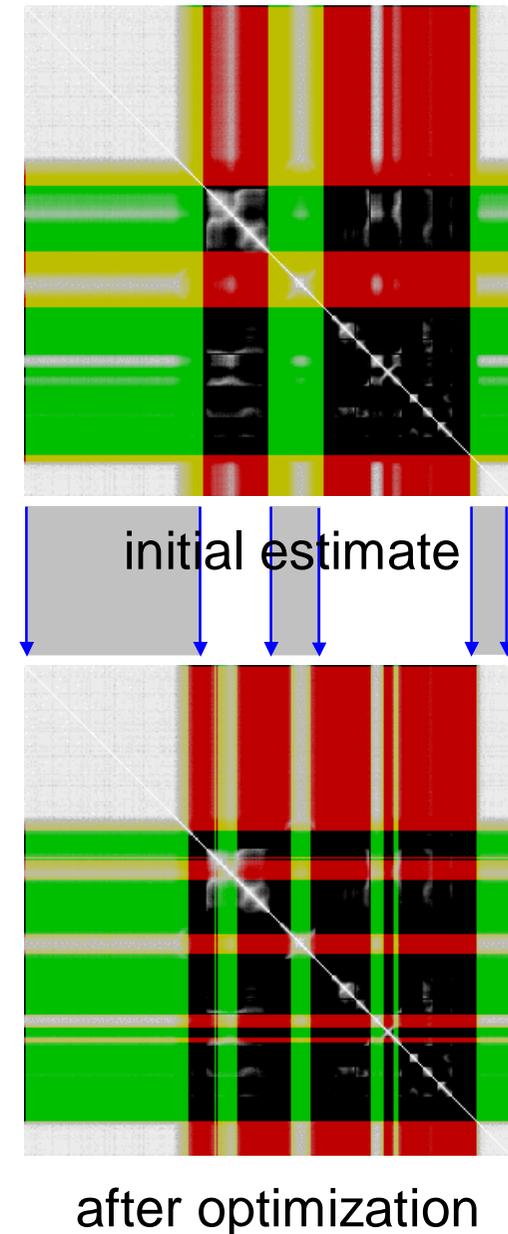
- Note that this is correct since M is symmetric and $M_{i;i} = 0$.
- Change of cost for removing k from T is obviously:

$$\Delta C_{-k} = -\Delta C_{+k}$$

- ➔ easy computation of cost difference for adding/removing single elements

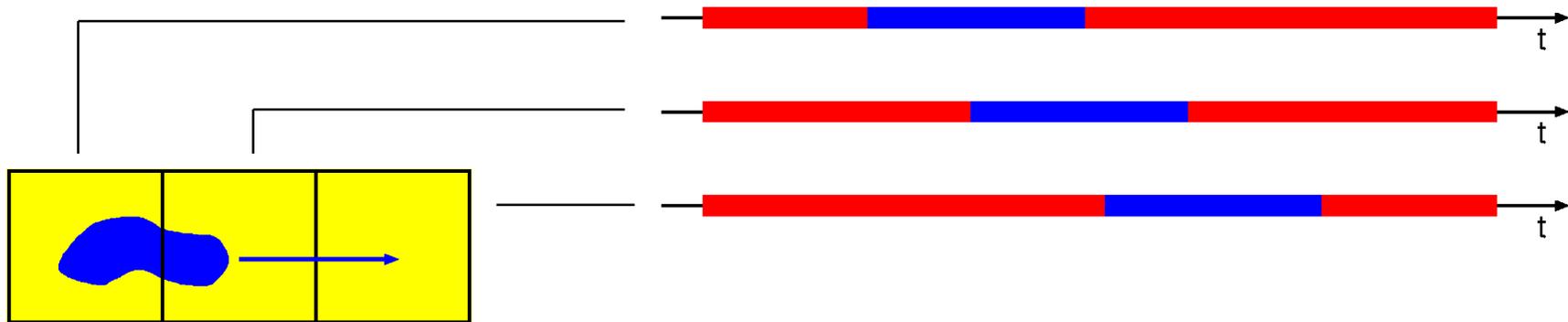
Decomposition Algorithm (2/2)

- Iterative optimization.
 - Start with initial estimate (e.g., over full time range)
 - For each frame k , check if adding/removing k from T decreases evaluation function.
 - If yes: add or remove
 - Repeat until convergence.
- Experiments show:
 - One optimization pass is sufficient.
- Depending on initialization, optimization converges to nearest local optimum.



Spatial Background-Period Prediction (1/2)

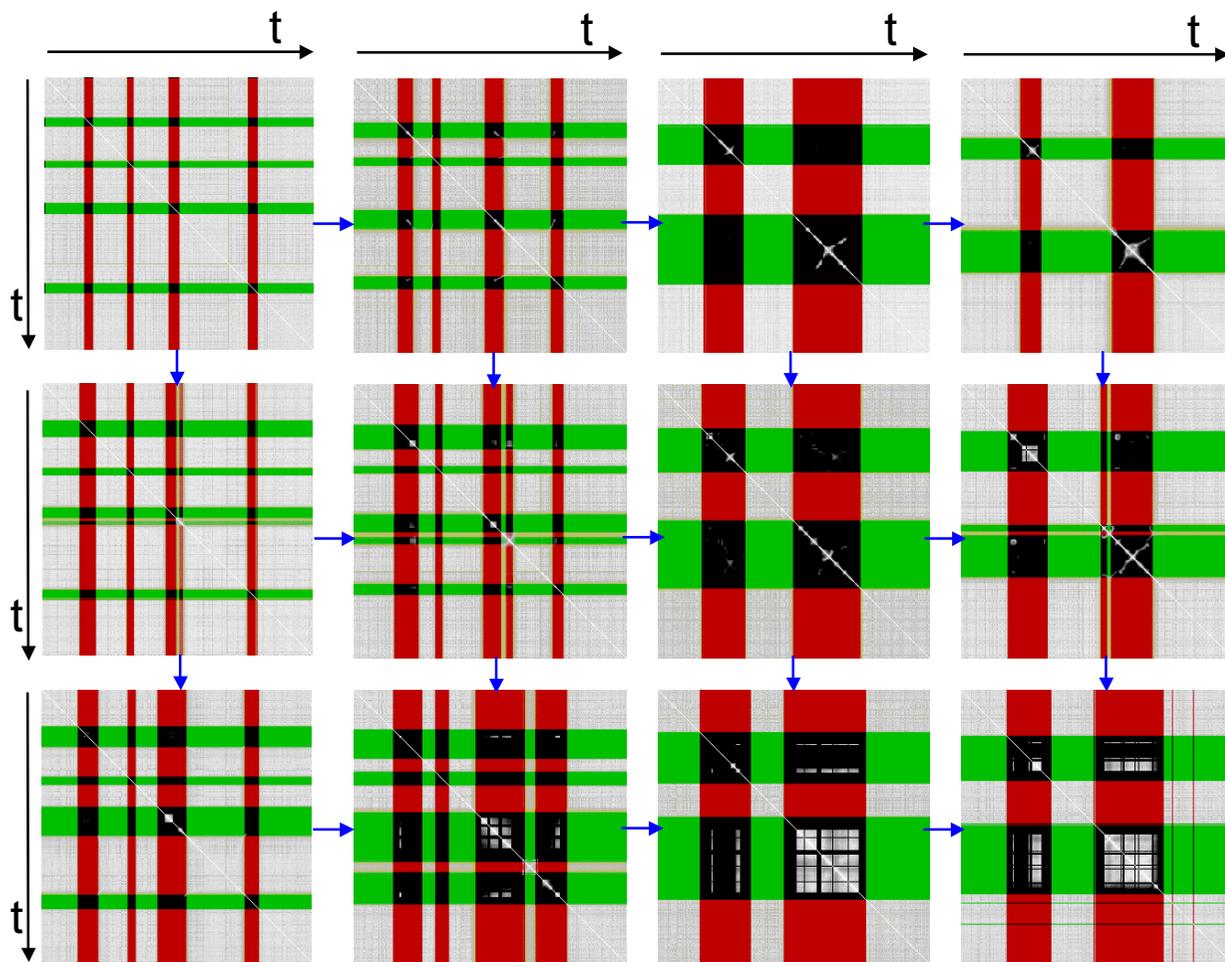
- There is a strong correlation of background times across neighboring blocks.



- Use solution of neighboring block as initialization of optimization.
- Two advantages:
 - faster convergence
 - solution of optimization is directed to the „correct“ local minimum.
Correct solution possible even if $t_{bkg} < t_{fgr}^i$.

Spatial Background-Period Prediction (2/2)

- Prediction can be extended to 2D prediction.
- Initialize as background if blocks above **and** to the left contain background.



Background-Image Reconstruction

- Block classification after matrix decomposition:



■ static background ■ moving foreground ■ static foreground

- Apply temporal median filter over blocks classified as **background**.

Results: Hall&Monitor / Highway



input sequence



result: median filter



result: our algorithm



input sequence



result: median filter



result: our algorithm

Results: Parking / University



input sequence



result: median filter



result: our algorithm



input sequence



result: median filter



result: our algorithm

Results: Highway with Snow / Urbicande



input sequence



result: median filter



result: our algorithm



input sequence



result: median filter



result: our algorithm

Results: Queue / VQEG-17



input sequence



result: median filter



result: our algorithm



input sequence
(from VQEG test set)



result: median filter



result: our algorithm

Conclusions

- New background image reconstruction algorithm
 - Classification of image blocks into three classes:
moving, static foreground, static background
 - Temporal median filter to combine static background blocks
- Very robust results, applicable even to difficult scenes.
- Low luminance bias; reconstructed background ideal for segmentation applications.
 - PSNR to ground-truth background (if successful)
 - median: 32 dB
 - our algorithm: 35 dB

(measured with *Hall & Monitor*, see paper)