

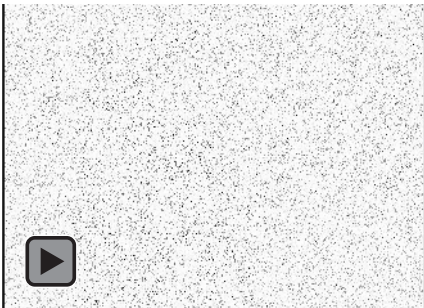
Video Reconstruction Problems

Final Presentation

Benjamin Busam, Philipp Jarde, Dennis Klöckner, Patricia Rachinger, Marco Ricci

Technische Universität München

July 14th, 2012



youtube, june 2012: "Dangerous Rites: Vine Bungee Jumping"

Contents

- 1 Problem Setting
- 2 Reconstruction with the ADM-Method
 - Theoretical Background
 - Implementation
 - Error Localization
- 3 Joint sparse and low rank matrix approximation
- 4 Computational Results

The Inpainting Problem

video inpainting: restore lost or corrupted pixels of a video.

→ needed for digitizing videos and removing errors.

possible errors:

- defects in old videos
- logos
- texts
- low resolution

Contents

- 1 Problem Setting
- 2 Reconstruction with the ADM-Method
 - Theoretical Background
 - Implementation
 - Error Localization
- 3 Joint sparse and low rank matrix approximation
- 4 Computational Results

The Model

Minimization Problem¹

$$\min_u \frac{\mu}{2} \| P u - f \|_2^2 + \varphi_{\text{reg}}(u) \quad (\text{P1})$$

with

u : reconstructed video

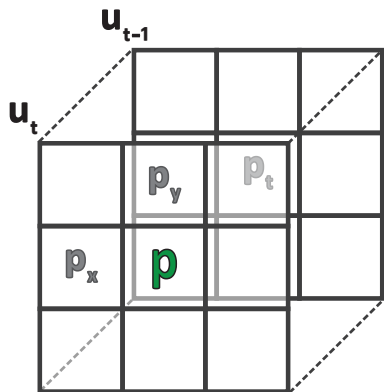
f : undamaged pixels in the corrupted video

P : projection onto undamaged pixels of the video

φ_{reg} : regularization function

¹Chan, Khoshabeh, Gibson, 2011

TV-Regularization



TV-Regularization

$$\varphi_{\text{reg}} = \sum_i \|D_i u\|_2$$

with

$$\|D_i u\|_2 = \left(\beta_x^2 [D_x u]_i^2 + \beta_y^2 [D_y u]_i^2 + \beta_t^2 [D_t u]_i^2 \right)^{\frac{1}{2}}$$

ADM I

1. transform (P1) to constrained problem:

$$\begin{aligned} \min_u \quad & \frac{\mu}{2} \|P u - f\|_2^2 + \sum_i \|w_i\|_2 \\ \text{s.t.} \quad & w_i = D_i u \quad \forall i \end{aligned}$$

2. form augmented Lagrangian

$$\begin{aligned} L_\alpha(w, u, \lambda) = & \frac{\mu}{2} \|P u - f\|_2^2 + \sum_i \|w_i\|_2 + \\ & \lambda_i^T (w_i - D_i u) + \frac{\alpha}{2} \|w_i - D_i u\|_2^2 \end{aligned}$$

ADM II

3. apply algorithm

Alternating Direction Method

FOR $k = 1, 2, \dots$ DO

1. $w^{k+1} = \arg \min_w L_\alpha(w, u^k, \lambda^k)$
2. $u^{k+1} = \arg \min_u L_\alpha(w^{k+1}, u, \lambda^k)$
3. $\lambda^{k+1} = \lambda^k - \gamma \alpha (w^{k+1} - Du^{k+1})$

END DO

According to the theory of Glowinski and Le Tallec (1989) this algorithm converges to the solution of (P1) if $0 < \gamma < \frac{1+\sqrt{5}}{2}\alpha$.

A closer look on the u-subproblem

u-subproblem: $u^{k+1} = \arg \min_u L_\alpha(w^{k+1}, u, \lambda^k)$

optimality condition: $\partial_u L_\alpha(w^{k+1}, u, \lambda^k) = 0$



$$Au = b^k$$

with:

- $A = \alpha D^T D + \mu P^T P$
- $b^k = \mu P^T f + D^T (\alpha w^{k+1} - \lambda^k)$

Result (video "jet"²)



Iterations	μ	β_x	β_y	β_t
2	10,000	1	1	0

²youtube, june 2012: "'Favorite' Jet Aircraft Takeoffs."

Problemsize for video "jet"

- resolution: 512 x 288
- duration: 6 sec
- frame rate: 25 fps

Problem size for video "jet"

- resolution: 512 x 288
- duration: 6 sec
- frame rate: 25 fps

⇒ 22,118,400 variables

Problemsize for video "jet"

- resolution: 512 x 288
- duration: 6 sec
- frame rate: 25 fps

Properties of A in video "jet"

- Size: 22,118,400 x 22,118,400
- Memory (sparse): 1.9 Gb
- Memory (full): 3913 Tb

Inexact solution of the u-subproblem

Problem: Matlab backslash operator produces "memory error".

Preconditioned conjugate gradient method

- $A = \alpha D^T D + \mu P^T P$ is symmetric and positive definite
- Incomplete Cholesky factorization as preconditioner
- Previous iteration as initial value
- PCG converges for video "jet" after 28, then 2 iterations with residuals $r < 10^{-10}$.

Overlapping Method



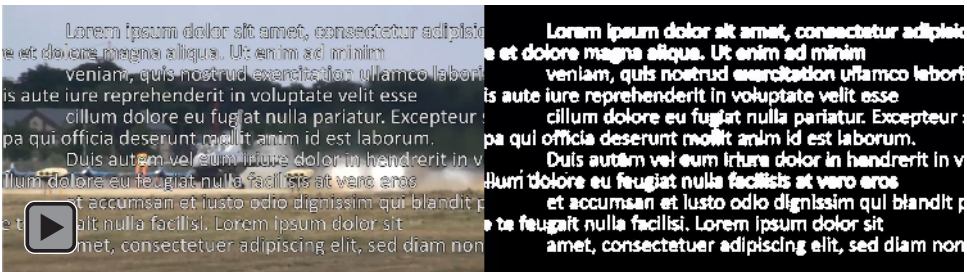
Difficulties

- Problem size and limited RAM capacity

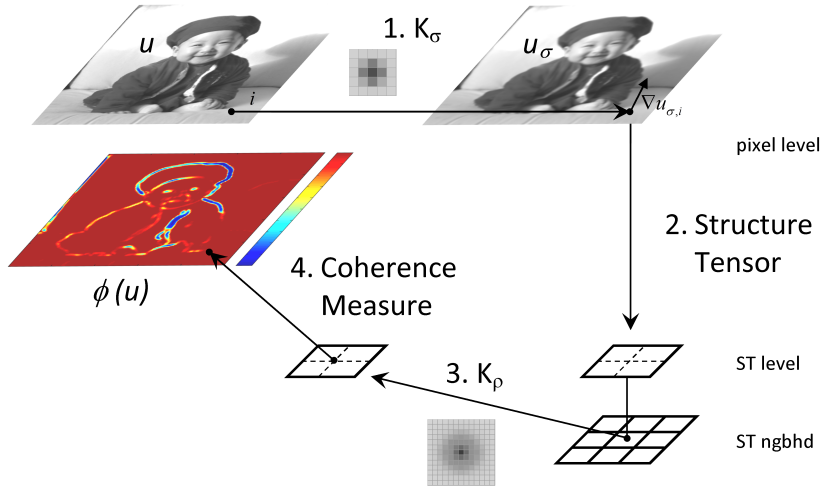
Solutions:

- Inexact solver (PCG) for the u-subproblem
- Overlapping method
- Corrupted pixels must be known in advance.

Corrupted Video & Dynamic Mask

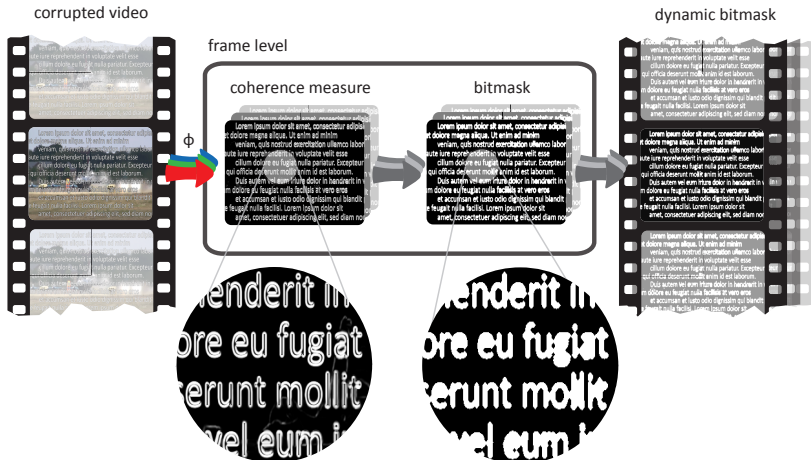


Coherence Measure³ ϕ

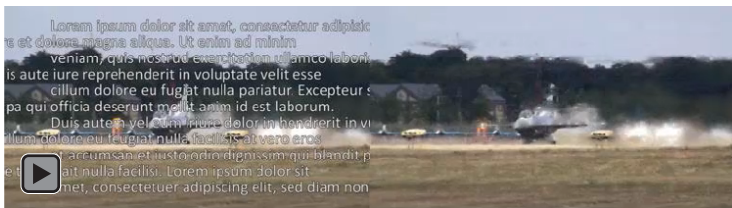


³Weickert, 1999. Picture: Wolfram Demonstrations Project, 2012.

Dynamic Mask



Corrupted Video & Reconstruction



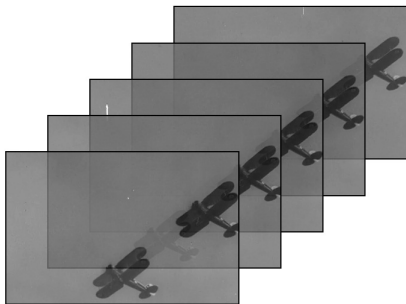
σ	ρ	calculation time for mask (1280x720)	scaling factor
0.1	1	approx. 12 $s/frame$ @ 1.6Ghz	0.3

Contents

- 1 Problem Setting
- 2 Reconstruction with the ADM-Method
 - Theoretical Background
 - Implementation
 - Error Localization
- 3 Joint sparse and low rank matrix approximation**
- 4 Computational Results

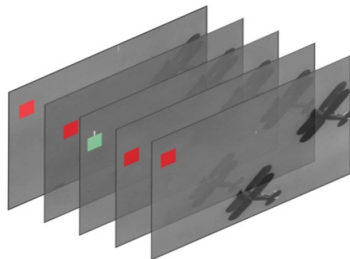
Idea

Same movie scene appears several times in neighbored frames.



⇒ use redundant information for detecting and correcting corrupted pixels

Patch matching



1. divide each frame into patches i
2. find similar patches i_k in neighbored frames
3. store column-wise similar patches i_k in a matrix P_i

Assumption

$$P_i = L_i + S_i$$

L_i : undamaged patches \rightarrow low rank matrix

S_i : corruption \rightarrow sparse matrix

The Model

Minimization Problem⁴

$$\min_{L_i, S_i} \|L_i\|_{\star} + \lambda \|S_i\|_1 \quad \text{s.t. } P_i = L_i + S_i \quad (\text{P2})$$

with $\|\cdot\|_{\star}$: nuclear norm: sum over all singular values

$\|\cdot\|_1$: sum of ℓ^1 -norms over all columns

λ : regularization parameter

Solving methods: APG or ADM

⁴Ji, Huang, Shen, Xu, 2011

Result⁵

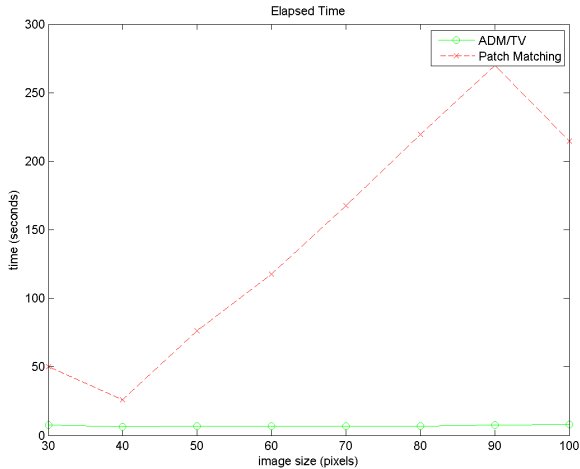
patch size	λ
16	1/16

⁵archive.org, june 2012: "Biplane Stunt Flying"

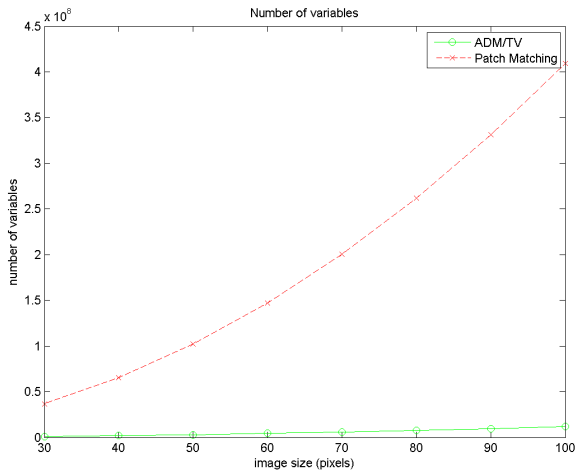
Contents

- 1 Problem Setting
- 2 Reconstruction with the ADM-Method
 - Theoretical Background
 - Implementation
 - Error Localization
- 3 Joint sparse and low rank matrix approximation
- 4 Computational Results**

Runtime



Memory Use



Conclusion

	ADM+TV	PM
short runtime	✓	×
low memory	✓	×
high damage	✓	×
preprocessing	✓	×

Outlook

- Speed up Patch Matching:
 - Parallel computation of patches
 - Run singular value decomposition on a GPU
- Combine ADM+TV and Patch Matching
- Different color models

Thank you for your attention!

Any questions?