Technion – IIT
Dept. of Electrical Engineering
Signal and Image Processing Lab

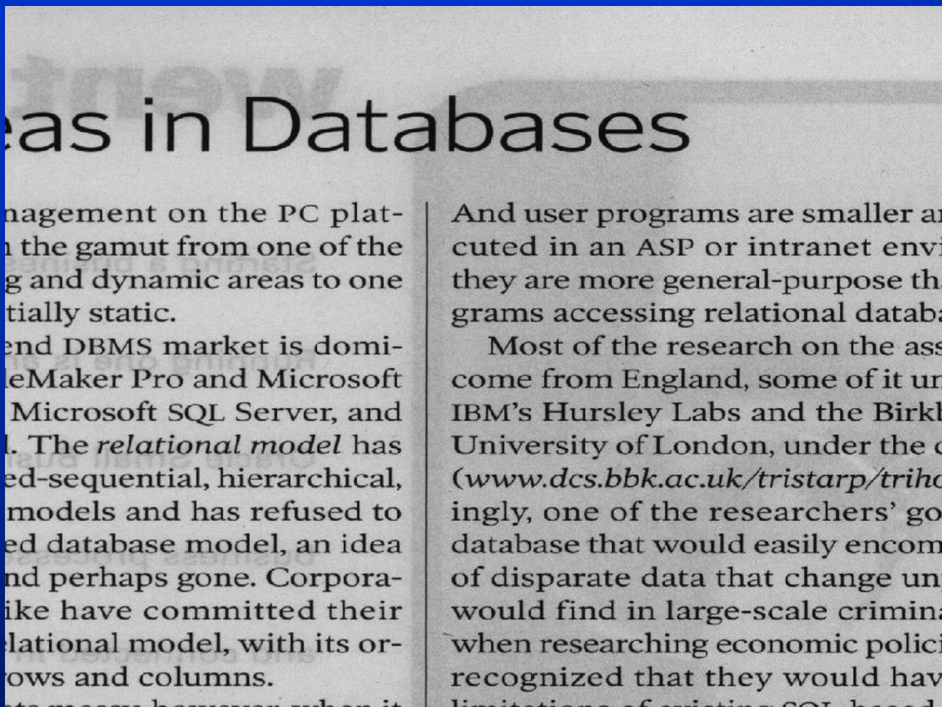
Show-Through Cancellation in Scanned Images

Boaz Ophir

M.Sc. Research under the supervision

of Prof. David Malah

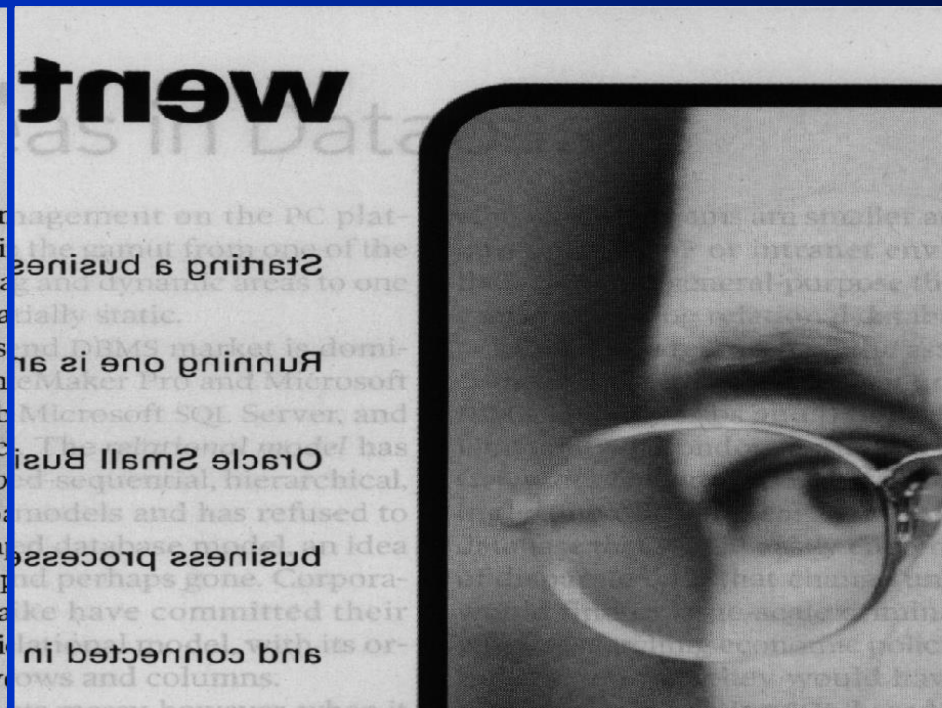
Show-Through in Scanned Images



eas in Databases

nagement on the PC plat-
n the gamut from one of the
g and dynamic areas to one
tially static.
end DBMS market is domi-
leMaker Pro and Microsoft
Microsoft SQL Server, and
l. The *relational model* has
ed-sequential, hierarchical,
models and has refused to
ed database model, an idea
nd perhaps gone. Corpora-
ike have committed their
lational model, with its or-
ows and columns.
ets messy however when it

And user programs are smaller ar
cuted in an ASP or intranet envi
they are more general-purpose tha
grams accessing relational databa
Most of the research on the ass
come from England, some of it un
IBM's Hursley Labs and the Birkb
University of London, under the c
(*www.dcs.bbk.ac.uk/tristarp/triho*
ingly, one of the researchers' go
database that would easily encom
of disparate data that change un
would find in large-scale crimina
when researching economic polici
recognized that they would hav
limitations of existing SQL-based



Goal : Remove cross-talk from images without distortion.

Talk Outline

- Previous work
- Physical model
- Decorrelation algorithm
  - Basic algorithm
  - Automatic background estimation
  - Local background normalization
  - Adaptive post-processing
  - Cascaded filter structure
- BSS algorithm
  - TV Regularization
  - Fidelity/Regularization Tradeoff
  - ICM
- Conclusions and Future work

Previous Work

Show-Through Removal in Scanned Images

- Sharma (`01)

Image Mixtures (Separating Reflections)

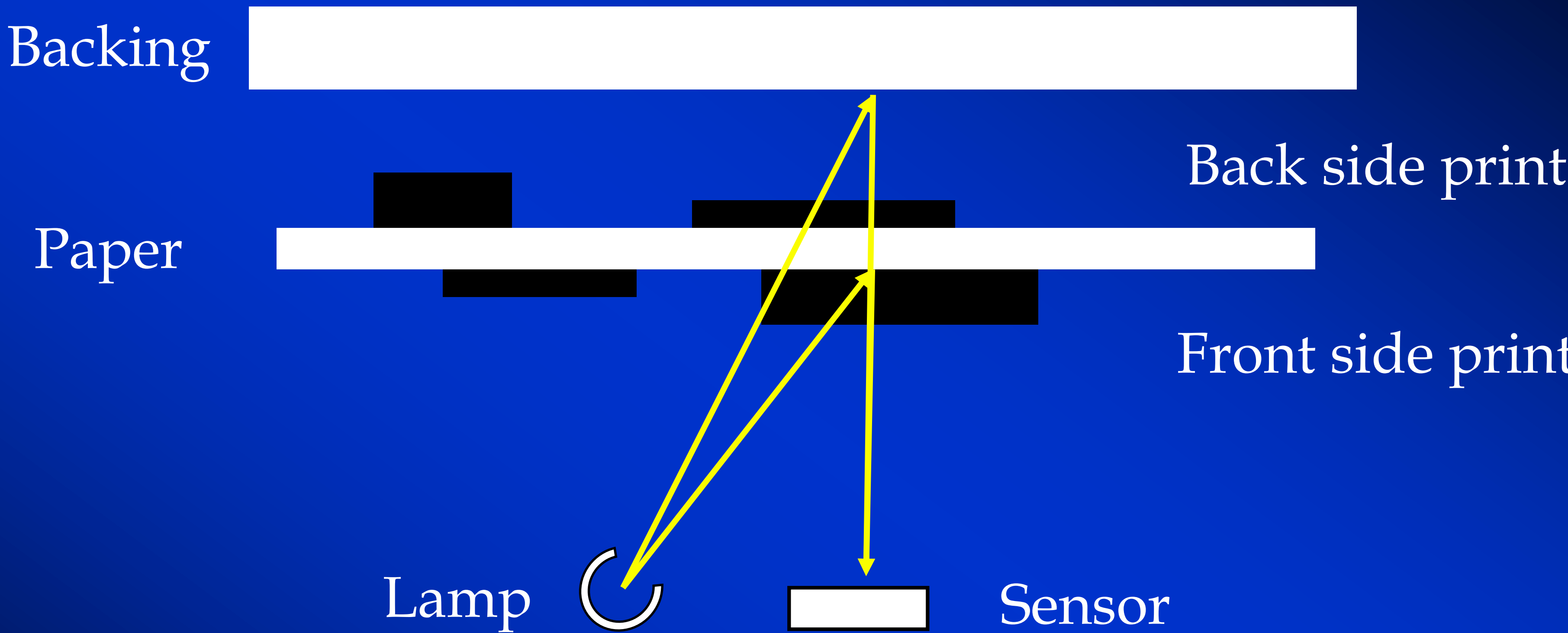
- Farid & Adelson (`99) – linear mixtures, ICA
- Schechner, Kiryati & Basri (`00) – convolutive mixtures, mutual information
- Bronstein^2, Zibulevsky & Zeevi (`03) – sparse ICA
- Sarel & Irani (`04) – spatially varying linear mixtures
- Levin, Zomet & Weiss (`04) – single image
- Tonazzini *et al* (`01-`05) – linear pointwise and convolutive image mixtures, MRFs

Related Problems

- Multi-channel systems : Weinstein, Feder & Oppenheim (`93), Thi & Jutten (`95) – BSS of convolutive mixtures
- Deconvolution : Bronstein^2, Zibulevsky & Zeevi (`05) - sparsity, Chan & Wong (`98) , Kaftory, Sochen & Zeevi (`05)– TV, many others…

Physical Model

Physical Model - 1



$$R_f^s = T_f^2(R_p + T_p^2 R_{bk} T_b^2)$$

$$R_f = T_f^2(R_p + T_p^2 R_{bk})$$

R - Reflectance

T - Transmittance

Physical Model - 2

Definitions:

Reflectance of white paper: $$R_p^w = R_p + T_p^2 R_{bk}$$

Normalized Optical Density

$$D = -\ln \frac{R}{R_p^w}$$

Normalized Optical Absorptance

$$A = 1 - \frac{R}{R_p^w}$$

Linearized Creation Model:

$$D_f^s(x,y) \simeq D_f(x,y) + \frac{T_p^2 R_{bk}}{R_p + T_p^2 R_{bk}} h(x,y) * A_b(x,y)$$

Show-Through Removal by Decorrelation

Basic Cleaning Algorithm
(Sharma 2001)

- R_p^w - Manual estimation.
- Front-side reflectance → Density.
- Back-side reflectance → Absorptance.
- For each pixel (progressing in a spatial contiguous order):
  - Compute show-through corrected density:

$$D_f(m,n) = D_f^s(m,n) - \sum_{k=-N}^{N} \sum_{l=-N}^{N} w(k,l) A_b^s(m-k, n-l)$$

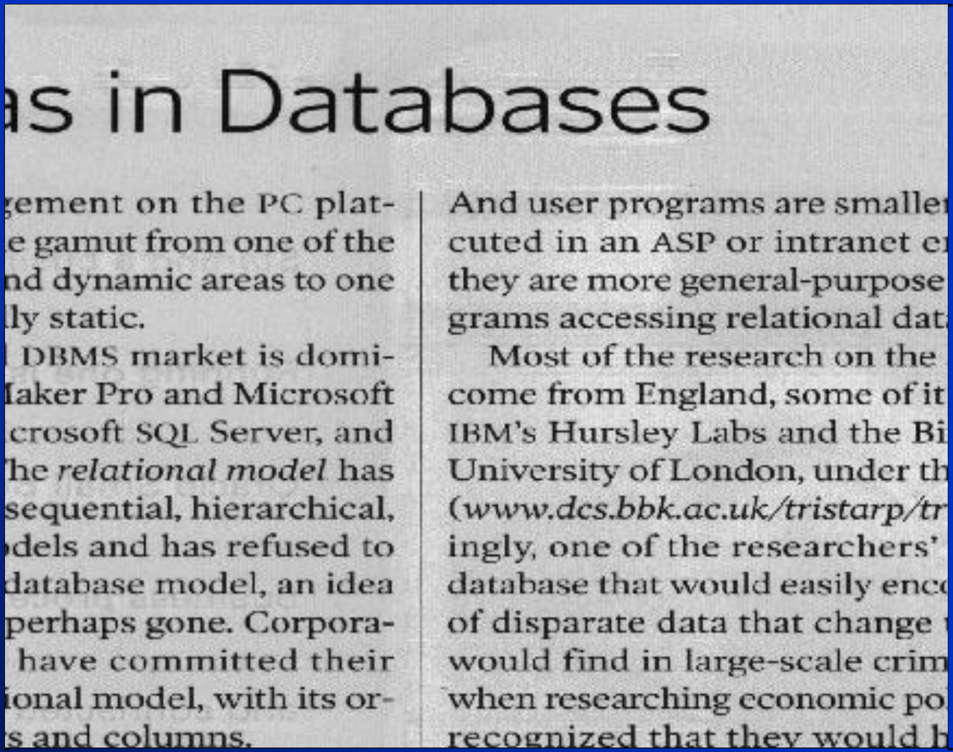
  - If back side has activity but not front side, update filter coefficients by LMS method:

$$w'(k,l) = w(k,l) + \mu D_f(m,n) A_b^s(m-k, n-l)$$

$$w(k,l) = w'(k,l)$$

  - Convert density to reflectance.

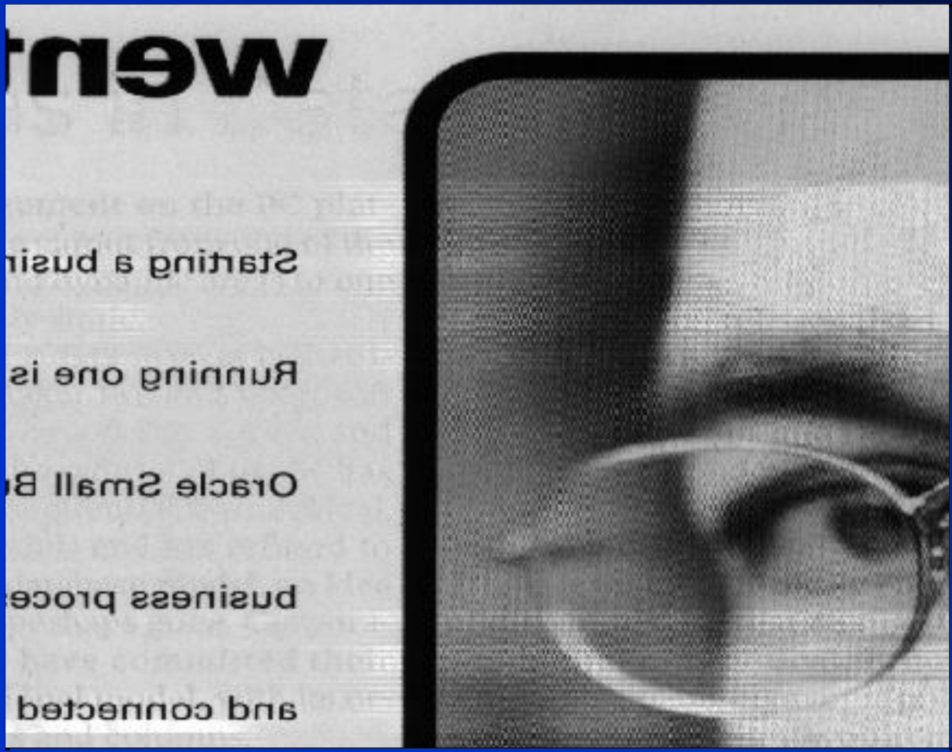
Results - Sharma



as in Databases

gement on the PC plat-
e gamut from one of the
nd dynamic areas to one
lly static.
l DBMS market is domi-
Maker Pro and Microsoft
crosoft SQL Server, and
The *relational model* has
sequential, hierarchical,
odels and has refused to
database model, an idea
perhaps gone. Corpora-
have committed their
ional model, with its or-
s and columns.

And user programs are smalle
cuted in an ASP or intranet e
they are more general-purpose
grams accessing relational dat
Most of the research on the
come from England, some of it
IBM's Hursley Labs and the Bi
University of London, under th
(*www.dcs.bbk.ac.uk/tristarp/tr*
ingly, one of the researchers'
database that would easily enc
of disparate data that change
would find in large-scale crim
when researching economic po
recognized that they would h

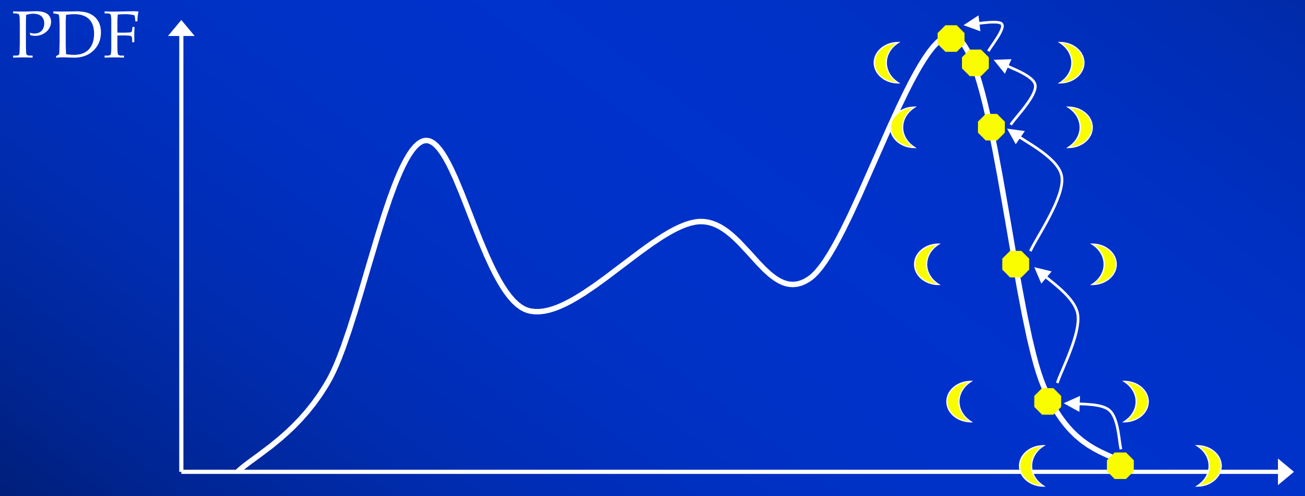


	Before	After
Corr	0.25	0.05
MI	0.1	0.02

Automatic Background Estimation

Mean-Shift algorithm (Fukunaga (`75), Cheng (`95), Meer (`02)) :

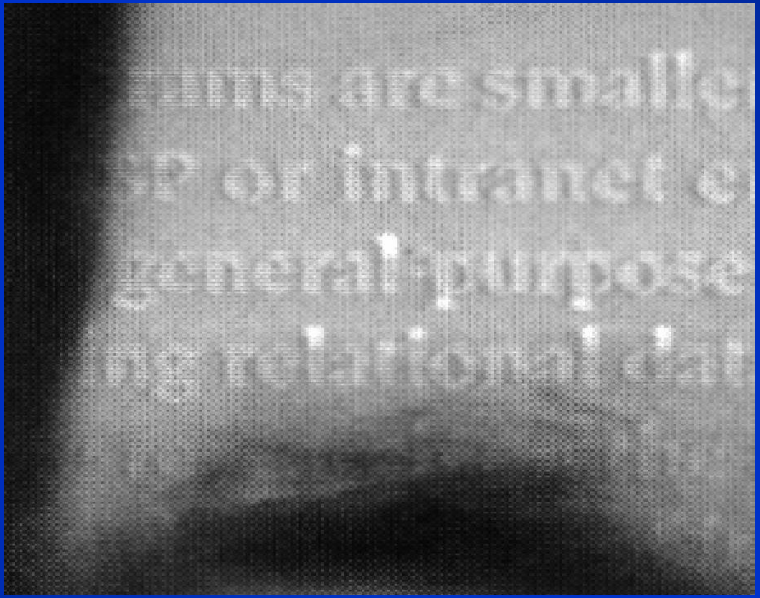
- Iterative steepest ascent algorithm
- Converges to local maximum



- Start point – brightest pixel
- Window size – proportional to signal STD

Local Background - 1

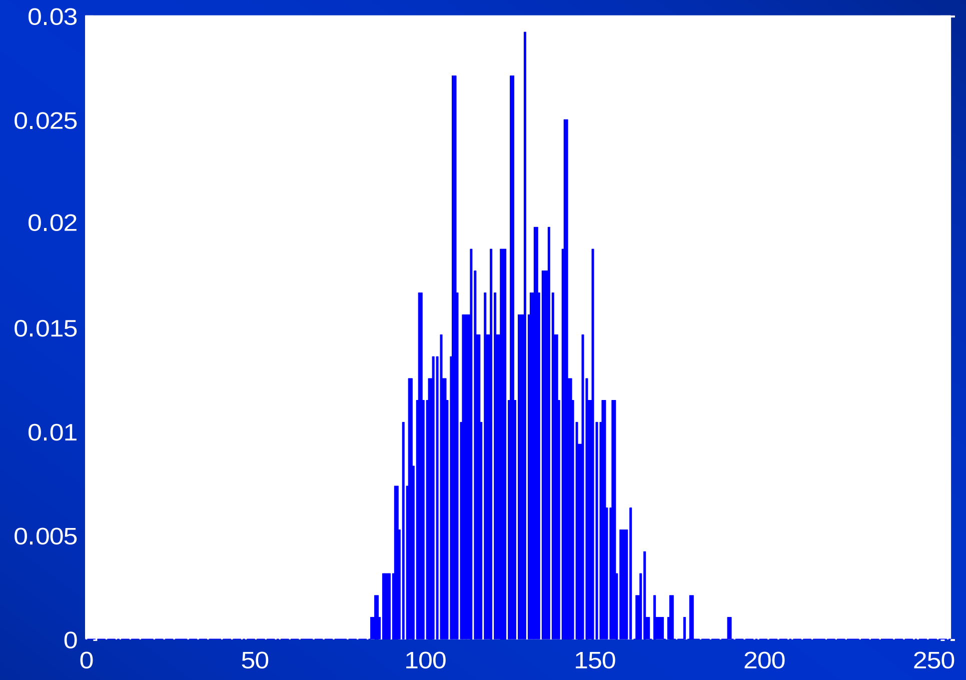
- Algorithm attempts to bring error to "zero", defined by R_p^w
- Problem:



- Solution : Local background is calculated by a mean shift process on histogram of local neighborhood

Local Background - 2

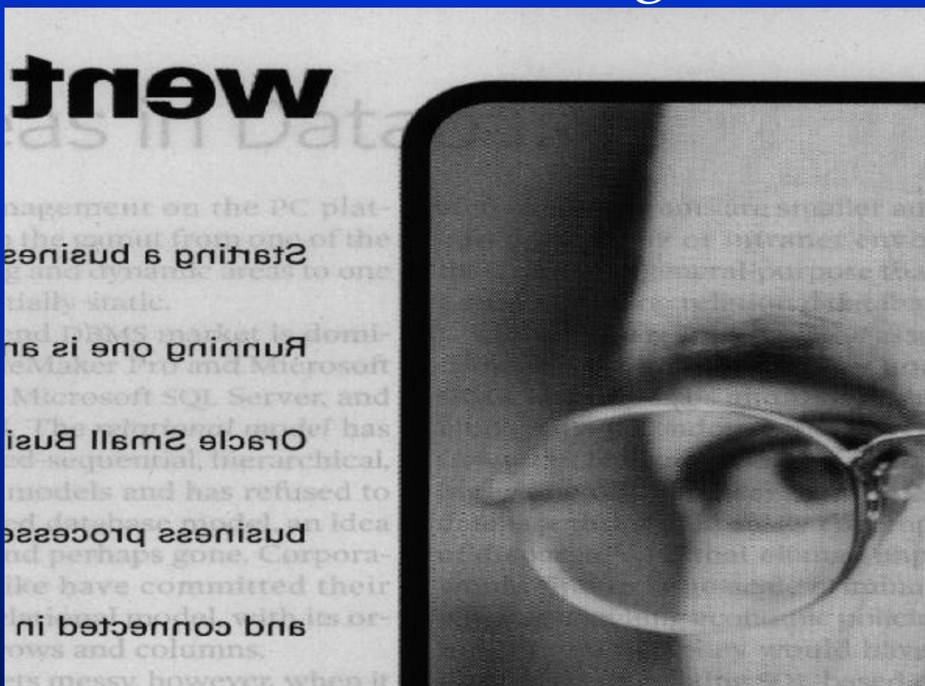
- Problem : Local histogram not a "good" approximation of PDF



- Solution : Adaptive mean-shift – variable window size
  - Increase window size until it covers enough pixels
- When reverse-side activity is high, Global value is used

Local Background - 3

Scanned Image



Background Image

- Global Background = 209

Decorrelation Systems

Construction System

$$\begin{cases} X_1 = S_1 + H_1 S_2 \\ X_2 = S_2 \end{cases}$$

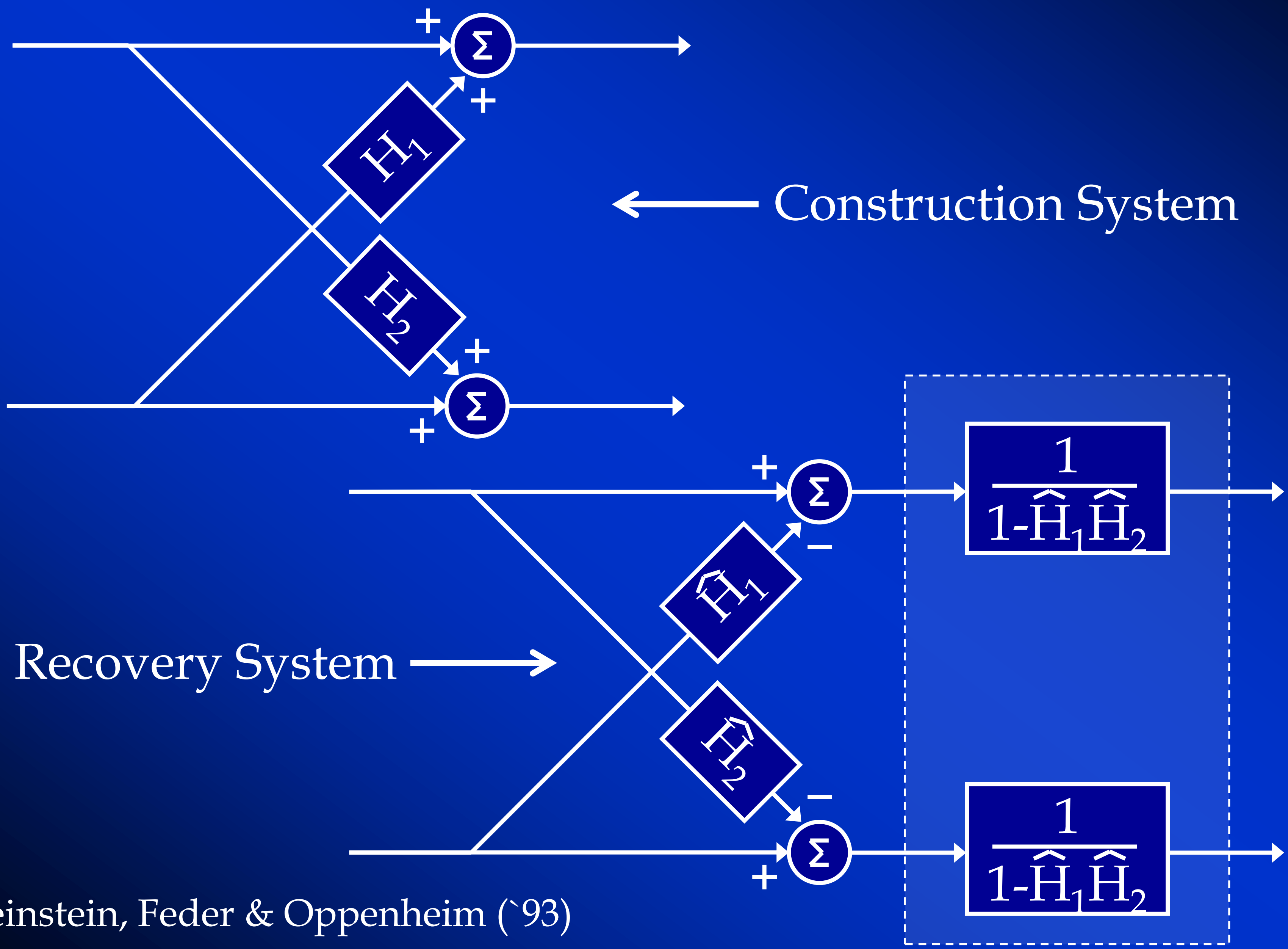
Recovery System

$$S_1 = X_2 - H_2 X_1$$

$$\begin{cases} X_1 = S_1 + H_1 S_2 \\ X_2 = S_2 + H_2 S_1 \end{cases}$$

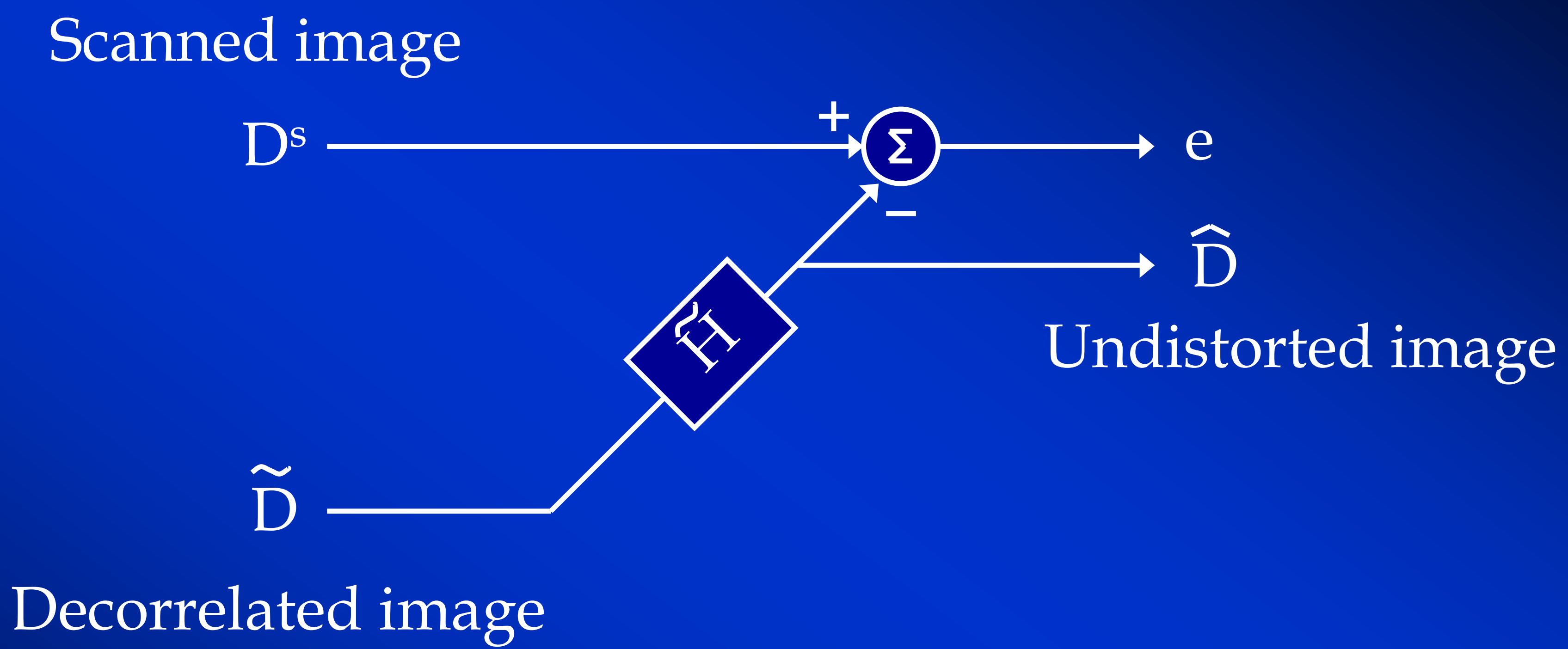
$$\begin{cases} S_1 = \dfrac{X_1 - H_1 X_2}{(1 - H_1 H_2)} \\ S_2 = \dfrac{X_2 - H_2 X_1}{(1 - H_1 H_2)} \end{cases}$$

Decorrelation Systems



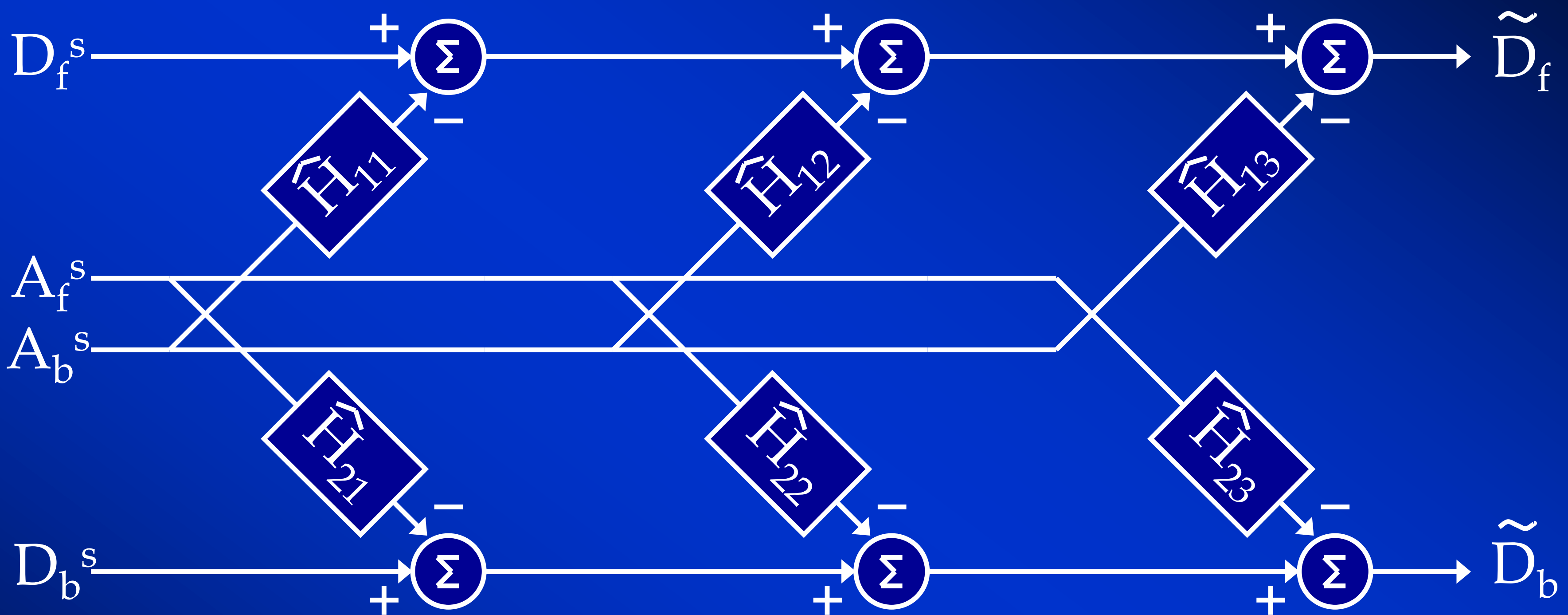
Weinstein, Feder & Oppenheim (`93)

Adaptive Post-Processing



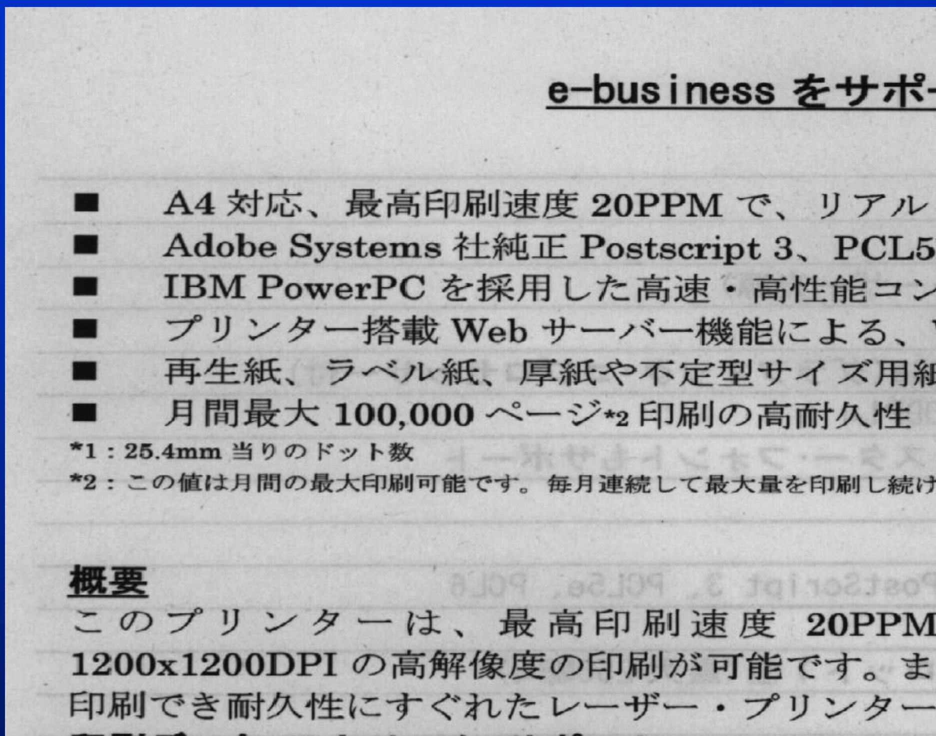
- Removes inherent distortion
- Removes some artifacts
- Degrades decorrelation somewhat

Cascaded Filter Structure



- Filter support increases with each stage
  - Typical values - 5x5, 9x9 & 15x15 pixels

Experimental Results

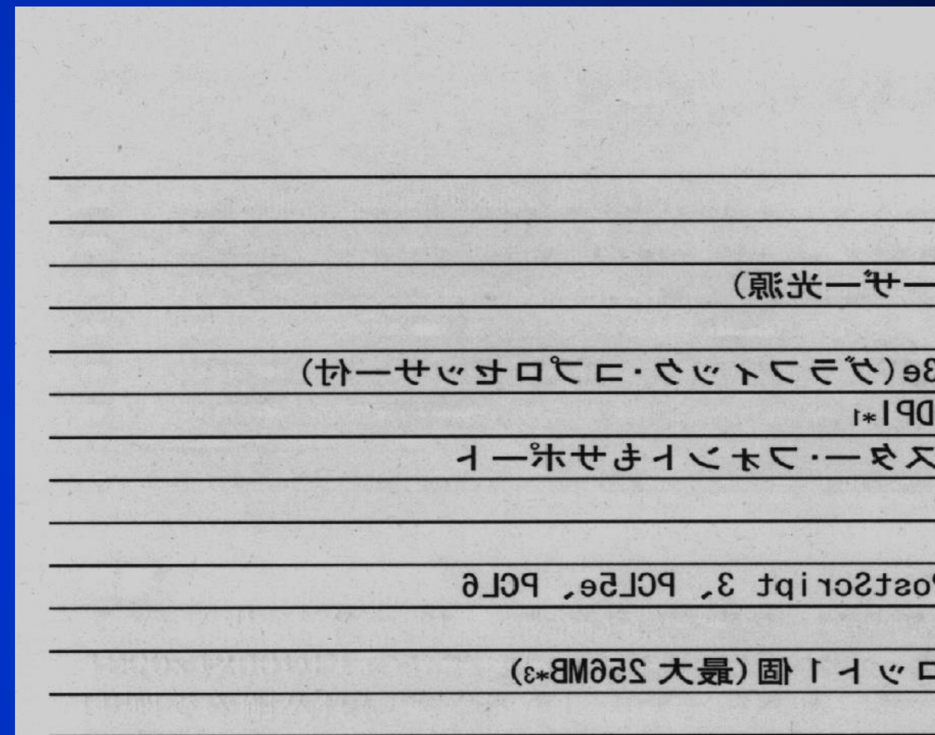


e-business をサポー

■ A4 対応、最高印刷速度 20PPM で、リアル
■ Adobe Systems 社純正 Postscript 3、PCL5
■ IBM PowerPC を採用した高速・高性能コン
■ プリンター搭載 Web サーバー機能による、W
■ 再生紙、ラベル紙、厚紙や不定型サイズ用紙
■ 月間最大 100,000 ページ*2 印刷の高耐久性

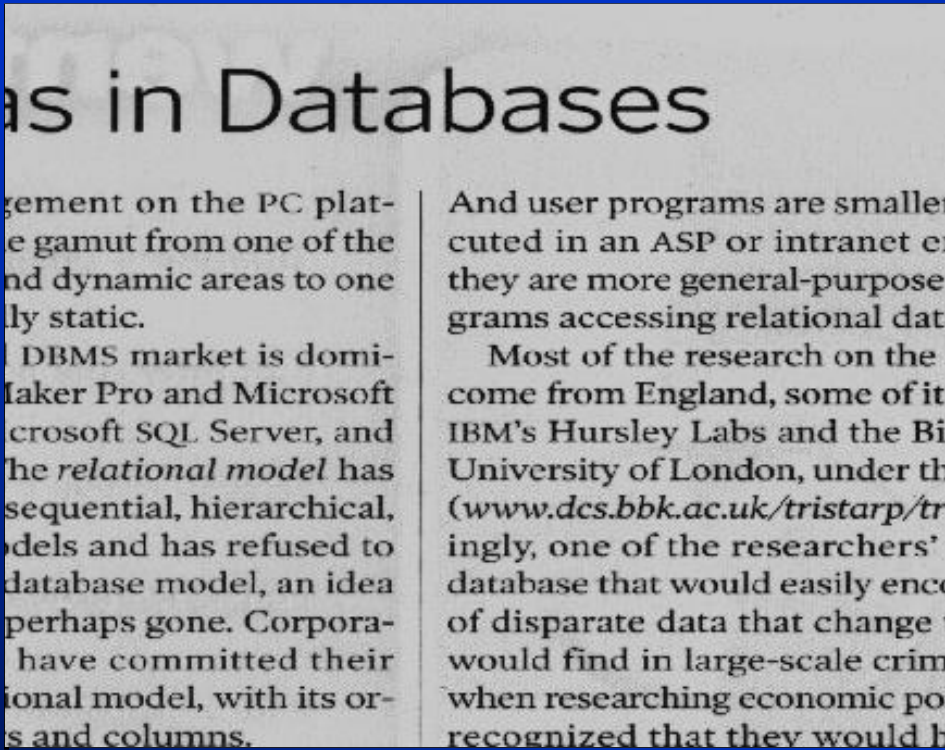
*1：25.4mm 当りのドット数
*2：この値は月間の最大印刷可能です。毎月連続して最大量を印刷し続け

概要
このプリンターは、最高印刷速度 20PPM
1200x1200DPI の高解像度の印刷が可能です。ま
印刷でき耐久性にすぐれたレーザー・プリンター



	Before	After
Corr	0.17	0.015
MI	0.09	0.025

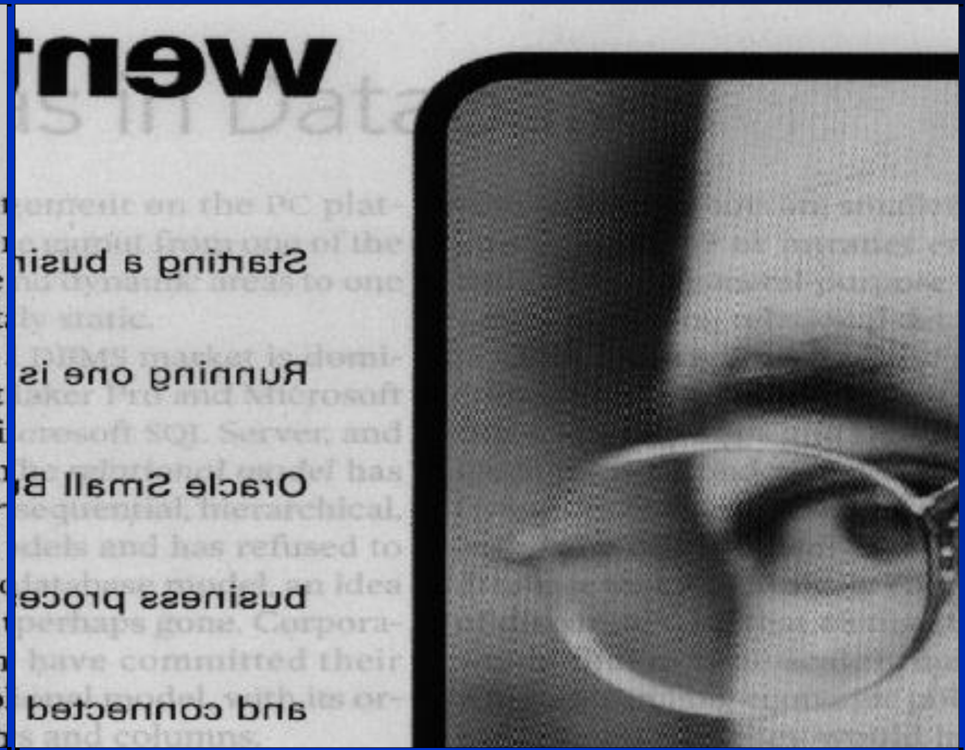
Experimental Results



as in Databases

gement on the PC plat-
e gamut from one of the
nd dynamic areas to one
lly static.
l DBMS market is domi-
laker Pro and Microsoft
crosoft SQL Server, and
The *relational model* has
sequential, hierarchical,
odels and has refused to
database model, an idea
perhaps gone. Corpora-
have committed their
ional model, with its or-
s and columns.

And user programs are smalle
cuted in an ASP or intranet e
they are more general-purpose
grams accessing relational dat
Most of the research on the
come from England, some of it
IBM's Hursley Labs and the Bi
University of London, under th
(*www.dcs.bbk.ac.uk/tristarp/tr*
ingly, one of the researchers'
database that would easily enc
of disparate data that change
would find in large-scale crim
when researching economic po
recognized that they would h



	Before	Sharma	After
Corr	0.25	0.05	0.03
MI	0.1	0.02	0.01

Decorrelation Algorithm Summary

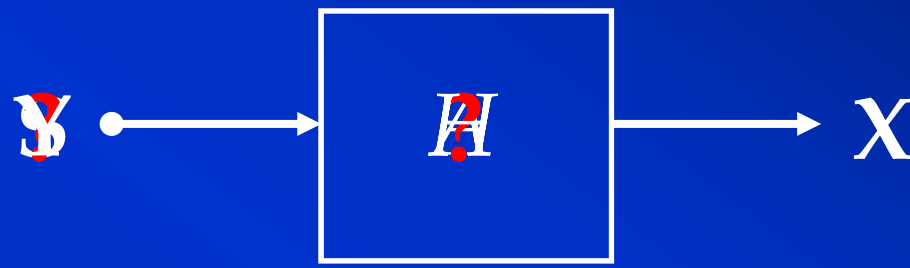
- Improved algorithm includes:
  - Local background normalization
  - Adaptive post-processing stage
  - Cascaded filter structure

- Algorithm achieves improved results.

- Algorithm handles scans of complex documents containing text and images.

Show-Through Removal
by
Blind Source Separation
(BSS)

Show-Through as a BSS Problem



- Goal – estimate images **Y** and mixing parameters *H*
- Method: Minimization of a Cost Function

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2$$

Image Separation as a Regularization Problem

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2 + \lambda_1 \operatorname{Re} g(Y_1) + \lambda_2 \operatorname{Re} g(Y_2)$$

- Assumption: Images Y1 and Y2 are independent
- Regularization Functions:
  - Smoothness
  - Markov Random Fields
  - Total Variation

Simple
Oversmoothing

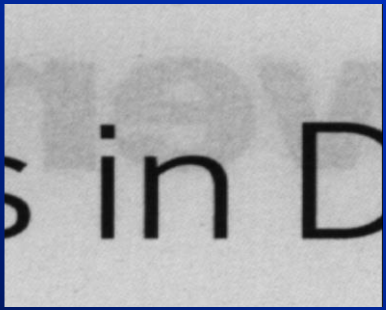
Edge Preserving
Many Parameters
Non Convex

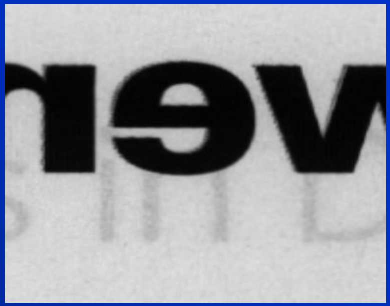
Edge Preserving
Convex

Fidelity/Regularization Tradeoff

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2 + \lambda_1 TV(Y_1) + \lambda_2 TV(Y_2)$$

- Tradeoff ratio λ determines strength of edges to be preserved.
- Strong *et al* (`97) suggest location dependent λ inversely proportional to gradient strength, in the context of image restoration.
  - λ is set separately for vertical and horizontal edges.
- Problem: Show-Through interference contains edges itself.





Fidelity/Regularization Tradeoff

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2 + \lambda_1 TV(Y_1) + \lambda_2 TV(Y_2)$$

- Solution: Weighting is set using both images.
- Strong edge in one image and weak edge at the same location in the other image => Probable Show Through scenario
  - Set **high** λ at show through edge
    - proportional to true (reverse side) edge strength
  - Set **low** λ at true edge
    - inversely proportional to edge strength

Example lambda maps

Scanned Images

Horizontal gradient
λ maps

Alternating Minimization

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2 + \lambda_1 TV(Y_1) + \lambda_2 TV(Y_2)$$

- Simultaneous minimization is very difficult:
  - Non-linear in joint space $\{H, \mathbf{Y}\}$
  - Non-convex in joint space $\{H, \mathbf{Y}\}$
  - Large number of variables
- Reduction method:

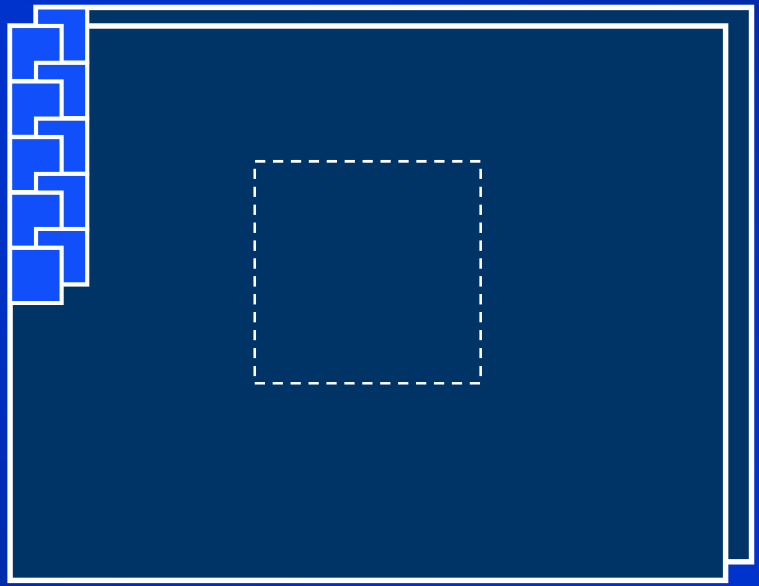
  Alternating Minimization in $\mathbf{Y}$ and H

  - Linear optimization problems
  - Convex cost functional

Iterated Conditional Modes (ICM)

Besag (1986)

- **Y** optimization is done on a very large number of dependent variables



- J is minimized one pixel location at a time.
- Each minimization is done only on partial sum of elements of J.
- Pixel values updated in place.
- Algorithm typically converges after 3-5 passes.
- To avoid directional preference we alternate pass direction.

Mixing Parameter Estimation

$$J = \left\| H(\mathbf{Y}) - \mathbf{X} \right\|_2^2$$

- Assumption: *H* is the same throughout the image

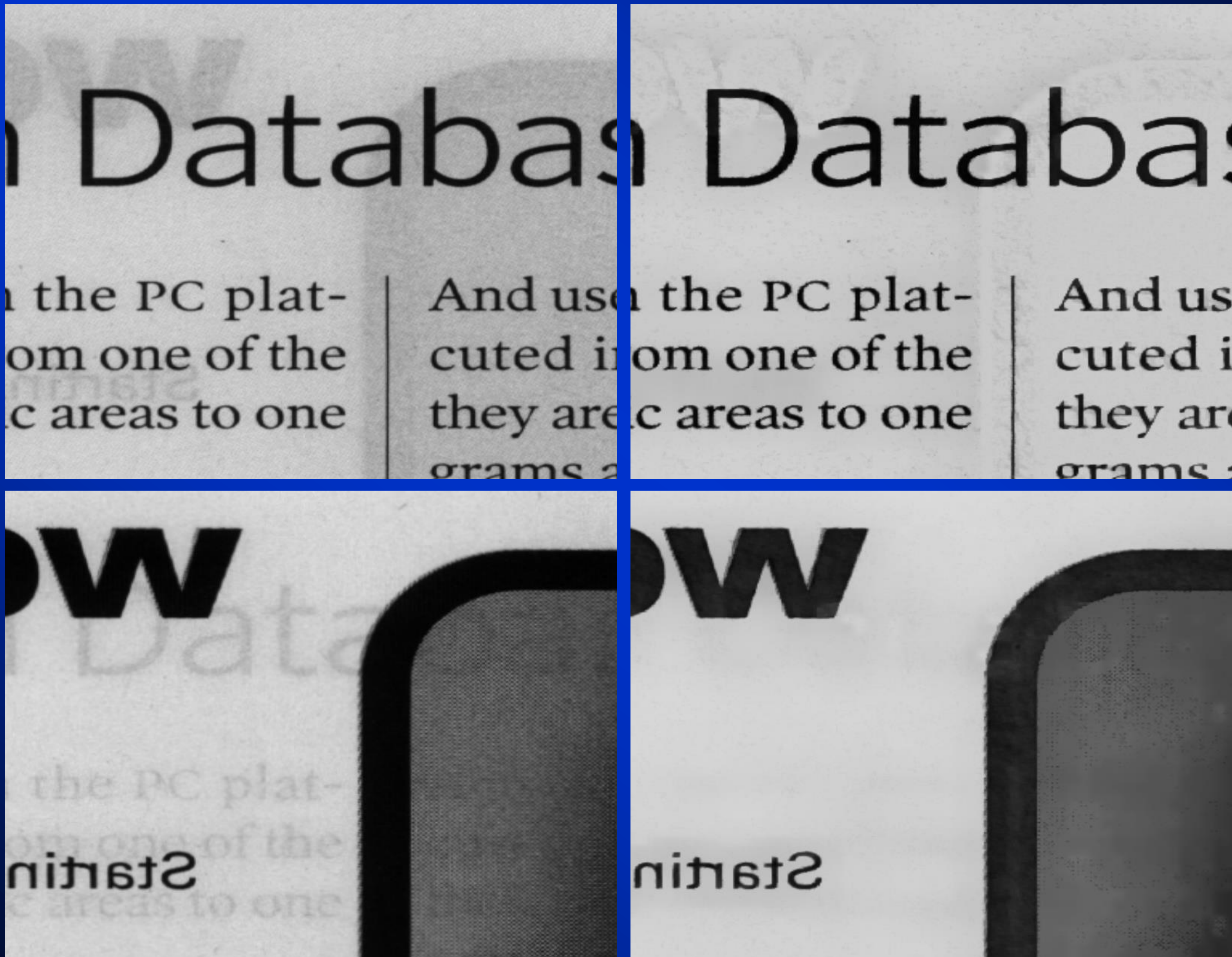
$$H = \begin{pmatrix} 1 & h_{12} * f(\cdot) \\ h_{21} * f(\cdot) & 1 \end{pmatrix}$$

- Least Squares problem
- Many pixels do not contribute to the solution
- Activity Masks – only pixels with single sided activity

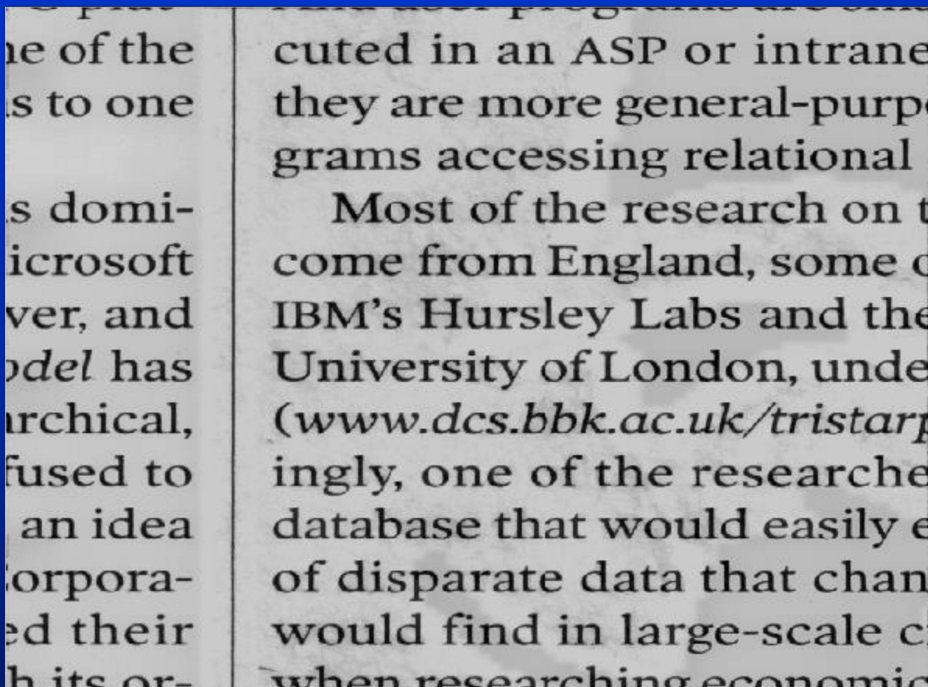
Minimization Algorithm

- Estimate Background
- Initialization : Y=X, $H = \begin{pmatrix} 1 & \varepsilon \\ \varepsilon & 1 \end{pmatrix}$
- Alternating Optimization:
  - Y Optimization:
    - Compute λ maps
    - ICM Iterations
      - Top-Bottom scan
      - Bottom-Top scan
  - *H* Optimization:
    - Compute activity masks
    - Least-Squares minimization
  - Normalize PSFs and images
- Repeat for larger filter support

Experimental Results



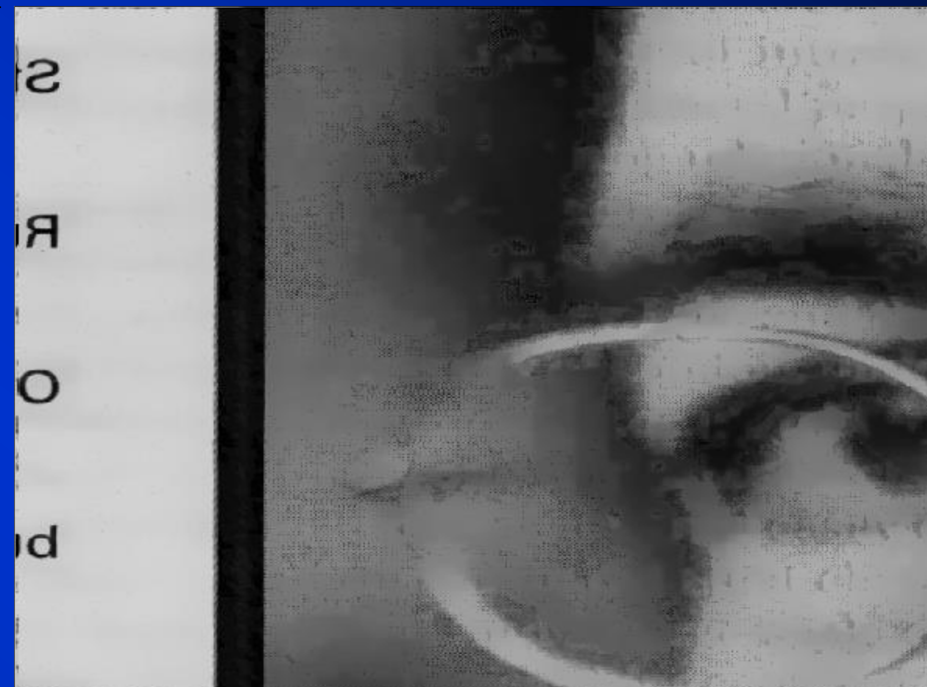
Experimental Results



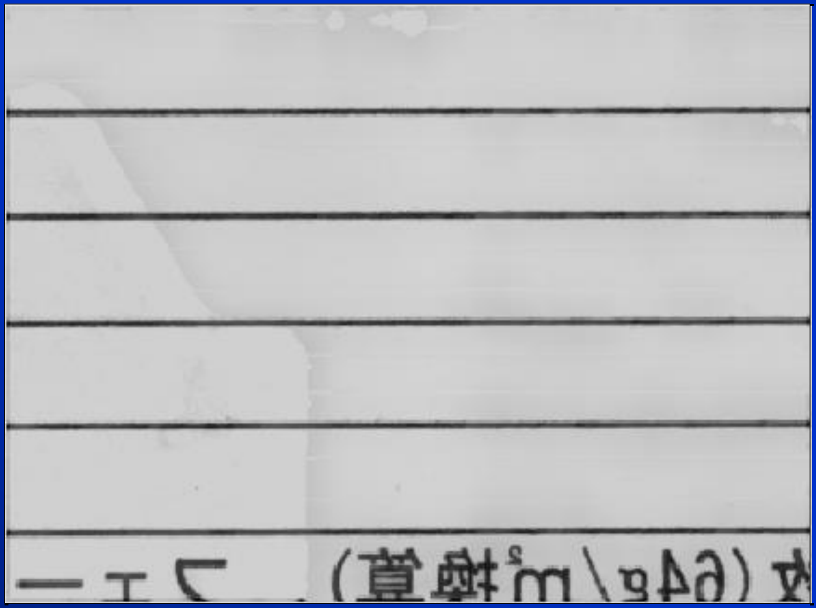
ne of the
s to one

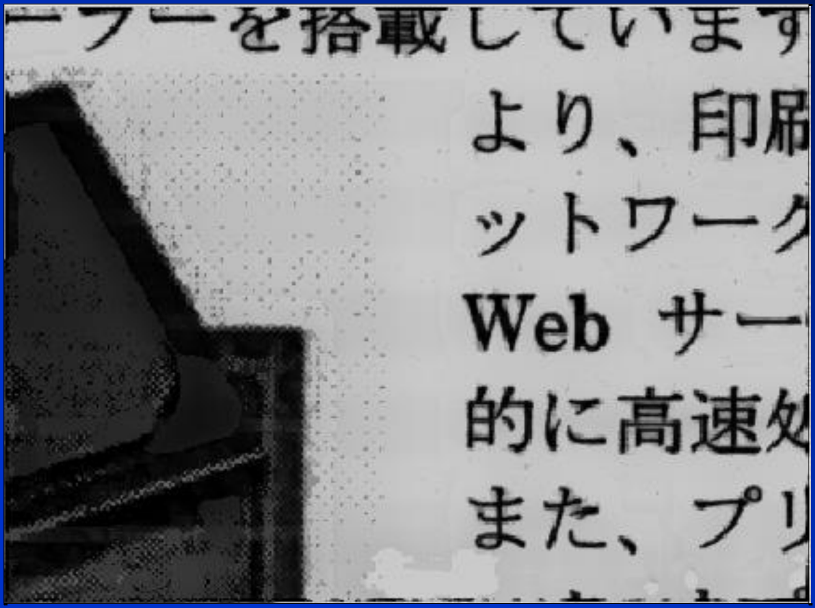
s domi-
icrosoft
ver, and
odel has
rchical,
fused to
an idea
orpora-
ed their
h its or-

cuted in an ASP or intrane
they are more general-purp
grams accessing relational
Most of the research on t
come from England, some o
IBM's Hursley Labs and the
University of London, unde
(*www.dcs.bbk.ac.uk/tristarp*
ingly, one of the researche
database that would easily e
of disparate data that chan
would find in large-scale c
when researching economic



Experimental Results





Conclusions and Future Work

Conclusions

- Show-Through as a non-linear convolutive mixture of images.
- Two algorithms were proposed:
  - Decorrelation – good results at low computation cost
  - BSS – high computation, way of the future

Future Work

- Technical Improvements
  - Better numerical approximations
  - Better optimization schemes
- Theoretical Extensions
  - Better image models
  - Color Images

Thank You

Algorithm Complexity

Decorrelation Algorithm:

- Per filtering stage
  - Per Pixel
    - Compute cleaned pixel value
    - Update PSF values

BSS Algorithm:

- Per ICM iteration
  - Per Pixel location
    - Minimization of partial sum of J
      - Number of elements in partial sum depends on PSF support
      - Optimization by Line-Search
        - Gradient estimation (2 function calculations)
        - Line-Search iterations (typically 4-6 steps)