



中国科学技术大学

University of Science and Technology of China

# 数学建模

## Mathematical Modeling

陈仁杰

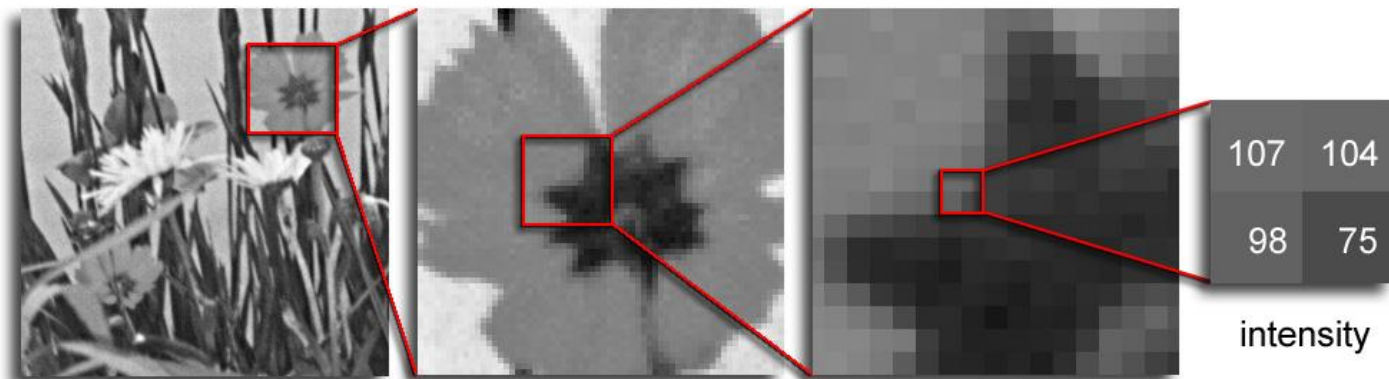
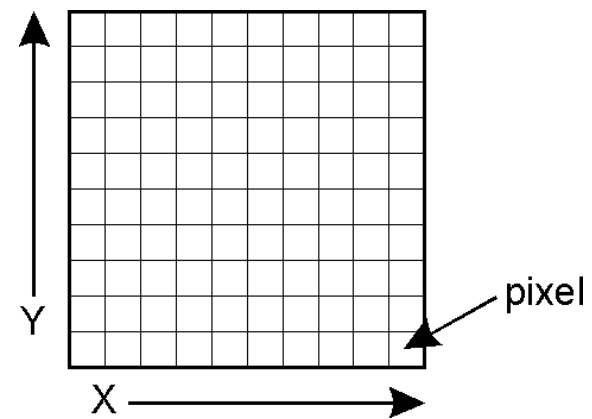
中国科学技术大学

# 矩阵分解模型

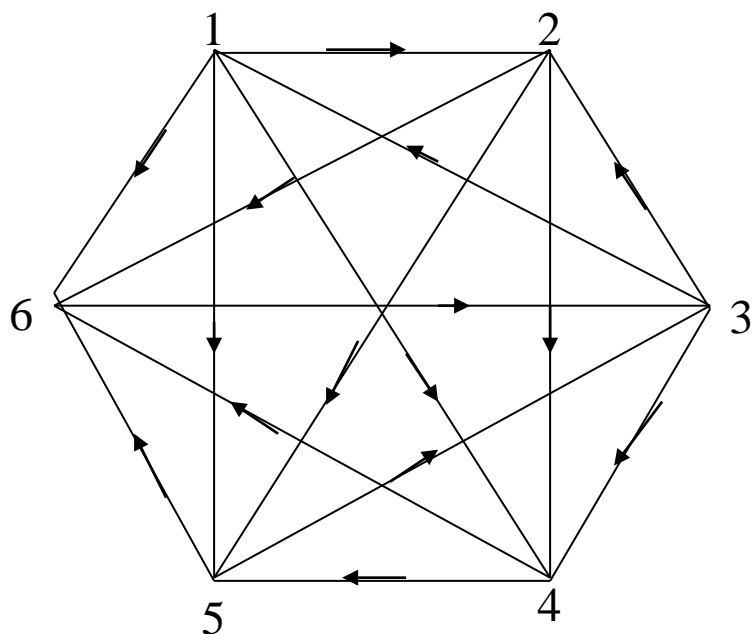
# 协方差矩阵

- 方差：单个随机变量的离散程度
- 协方差：两个随机变量的相似程度
- 协方差矩阵

# 光栅图像：矩阵



# 图的邻接矩阵



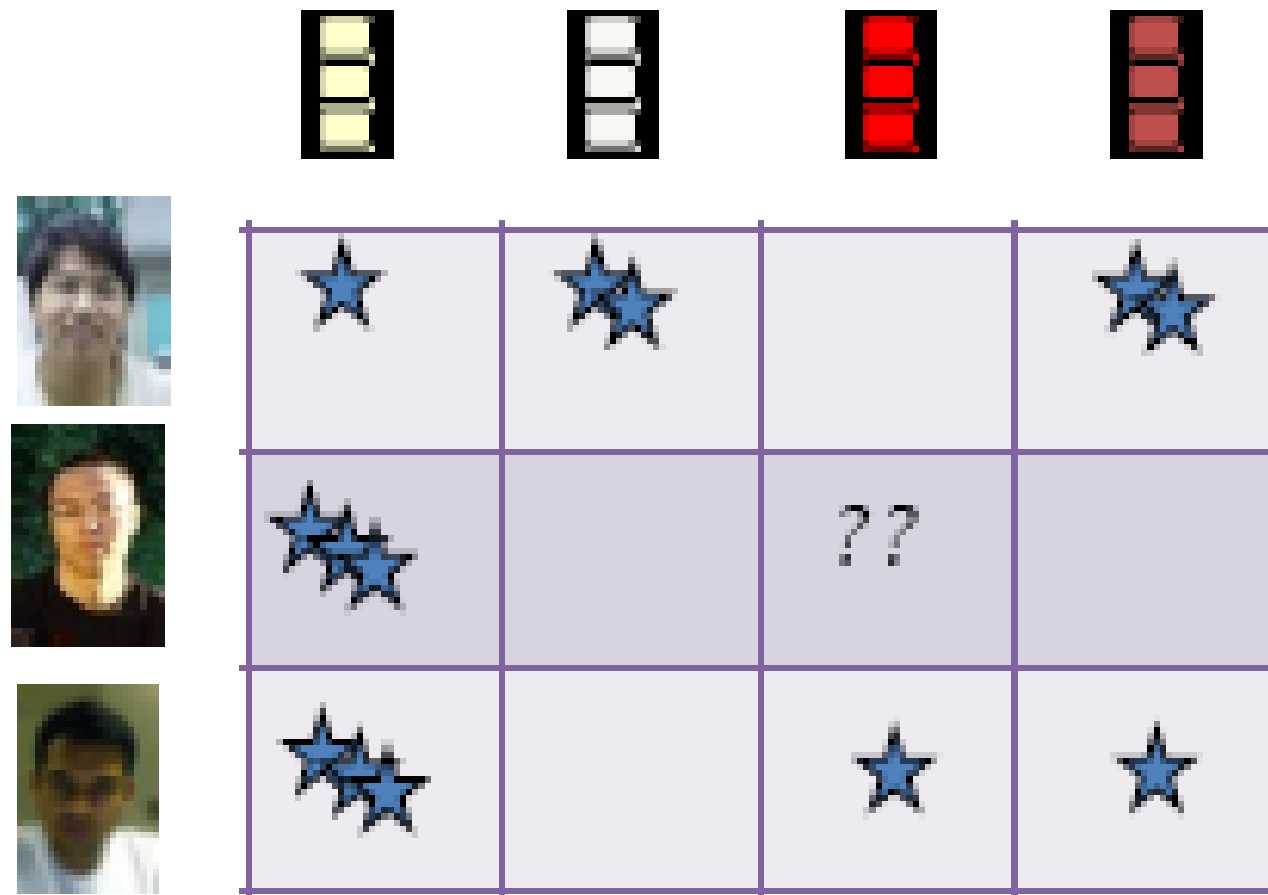
$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

# 问题： 用户-物品评分

User /item	1	2	3	4	5	.....
1	5	4	4.5	?	3.9	.....
2	?	4.5	?	4.5	?	.....
3	4.5	?	4.4	4	4	.....
4	?	4.8	?	?	4.5	.....
5	4	?	4.5	5	?	.....
.....	.....	.....	.....	.....	.....	.....

<http://blog.csdn.net/zhongkejingwang>

# Netflix Challenge



# 理解矩阵

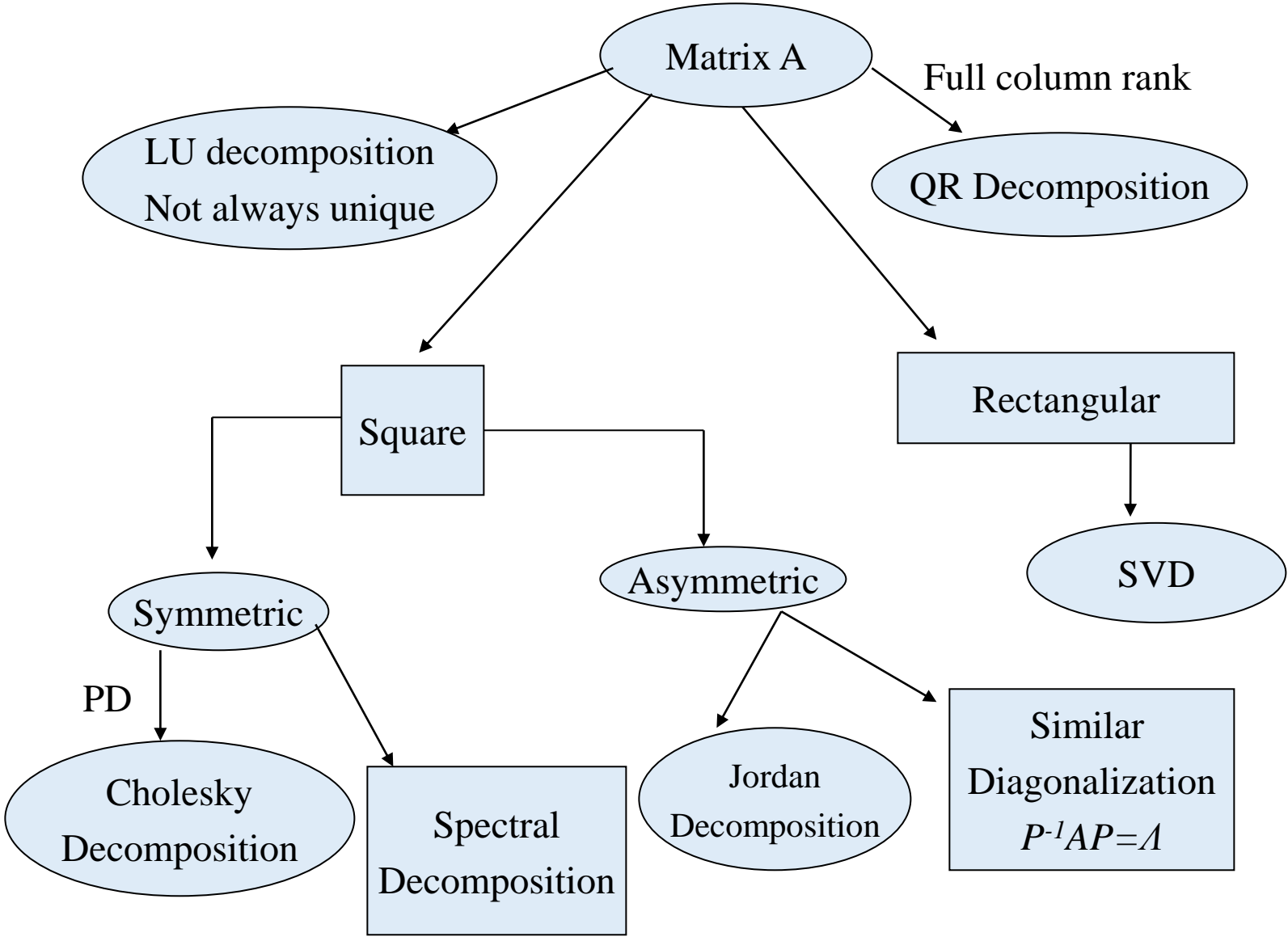
- 变换
- 线性变换
- 基函数的转换



# Matrix Decomposition (Factorization)

- LU decomposition
- QR decomposition
- Cholesky decomposition
- Jordan Decomposition
- Spectral decomposition (Eigendecomposition)
- Singular value decomposition (SVD)
- Low rank decomposition
- ...

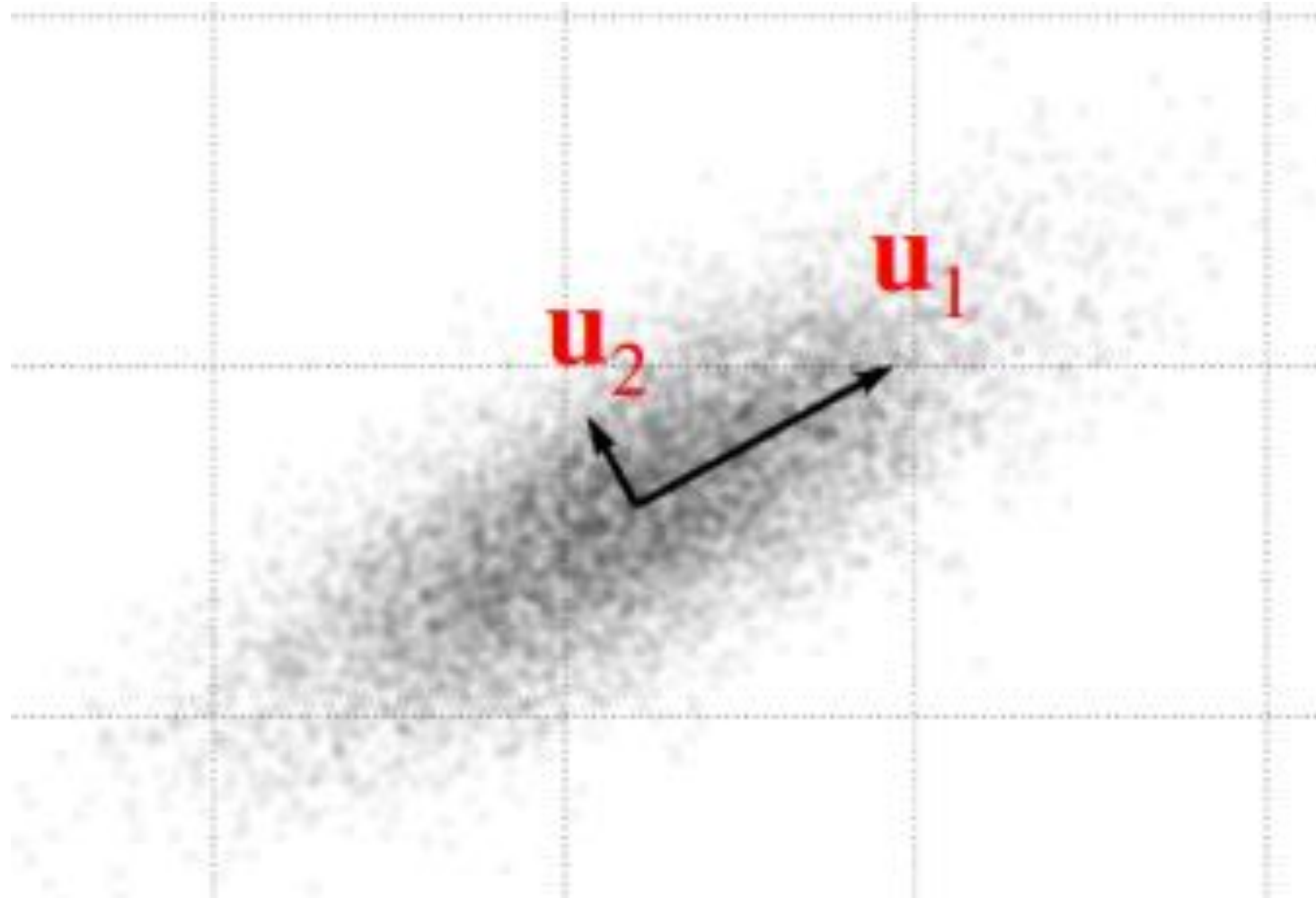
# 矩阵分解

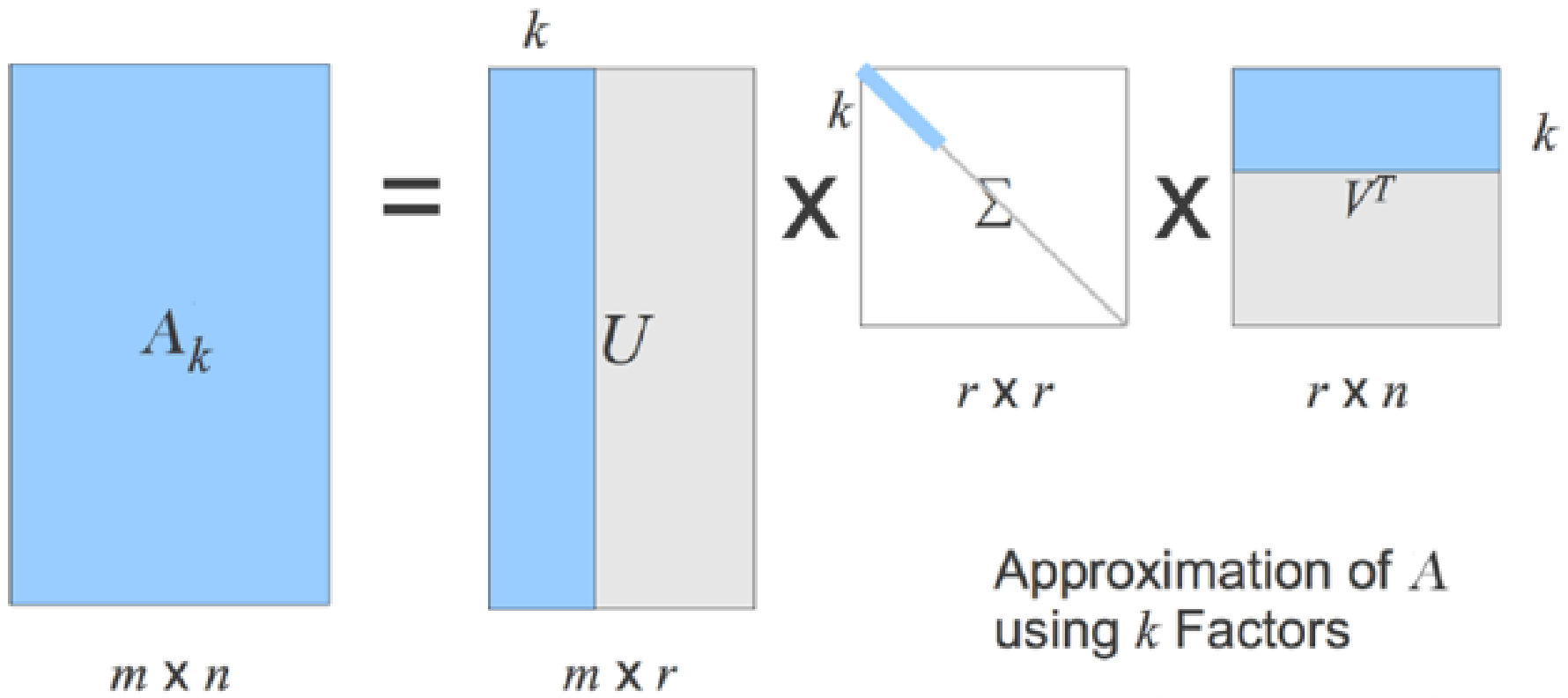


# SVD

图像压缩

# SVD



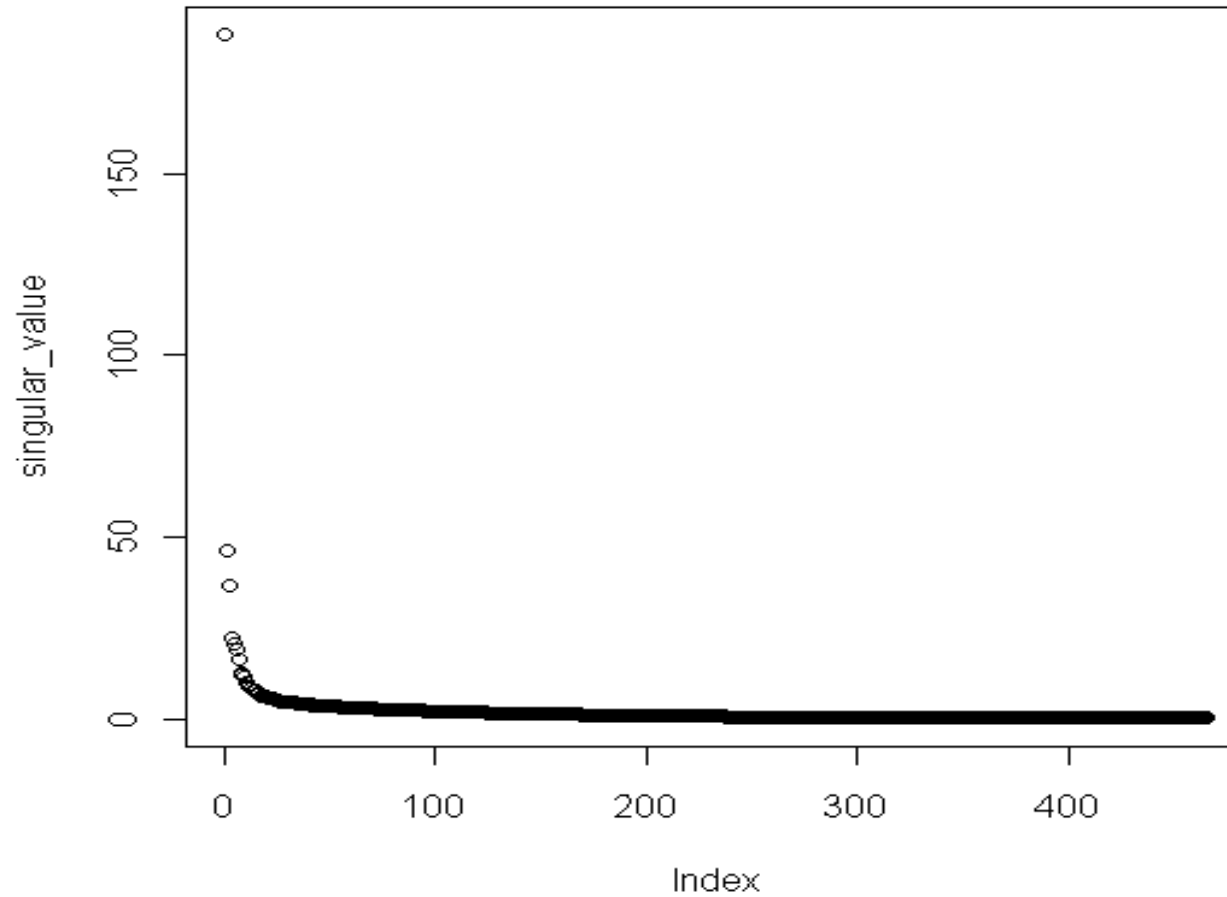


# Image Compression Example

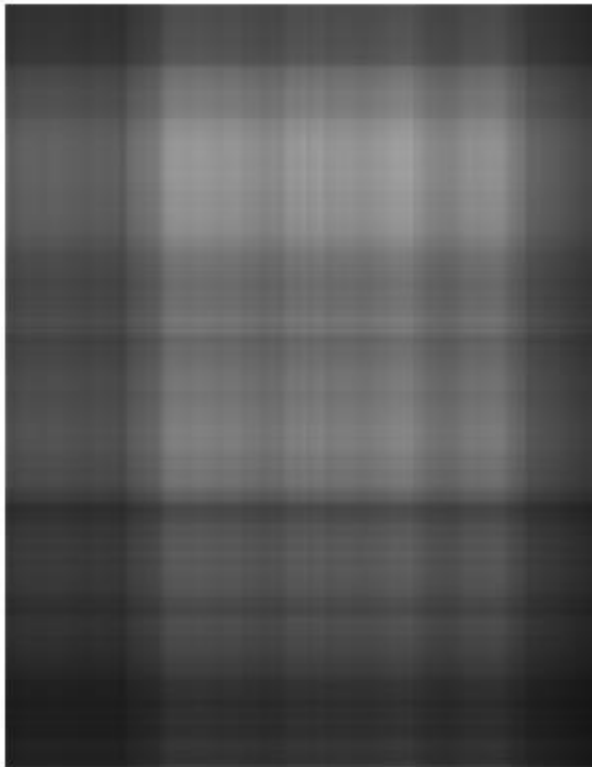


This image is  $600 \times 465$  pixels

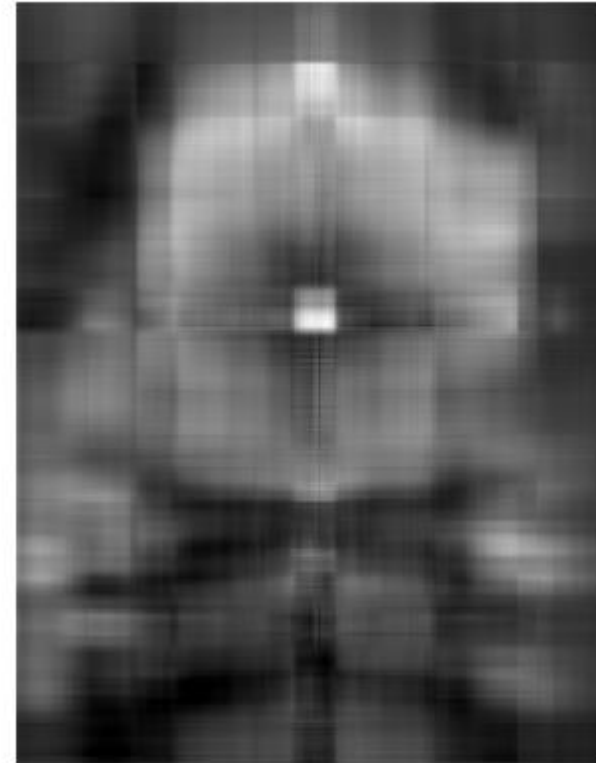
# Singular values of flowers image



# Low rank Approximation to flowers image



Rank-1 approximation



Rank- 5  
approximation



# Low rank Approximation to flowers image



Rank-20 approximation



Rank-30 approximation

# Low rank Approximation to flowers image



Rank-50 approximation



Rank-80 approximation

# Low rank Approximation to flowers image



Rank-100 approximation



Rank-120 approximation

# Low rank Approximation to flowers image



Rank-150 approximation



True Image

# 推荐系统

# 用户-物品评分

User /item	1	2	3	4	5	.....
1	5	4	4.5	?	3.9	.....
2	?	4.5	?	4.5	?	.....
3	4.5	?	4.4	4	4	.....
4	?	4.8	?	?	4.5	.....
5	4	?	4.5	5	?	.....
.....	.....	.....	.....	.....	.....	.....

<http://blog.csdn.net/zhongkejingwang>

	item 1	item 2	item 3	item 4
user 1	R11	R12	R13	R14
user 2	R21	R22	R23	R24
user 3	R31	R32	R33	R34

**R**

=

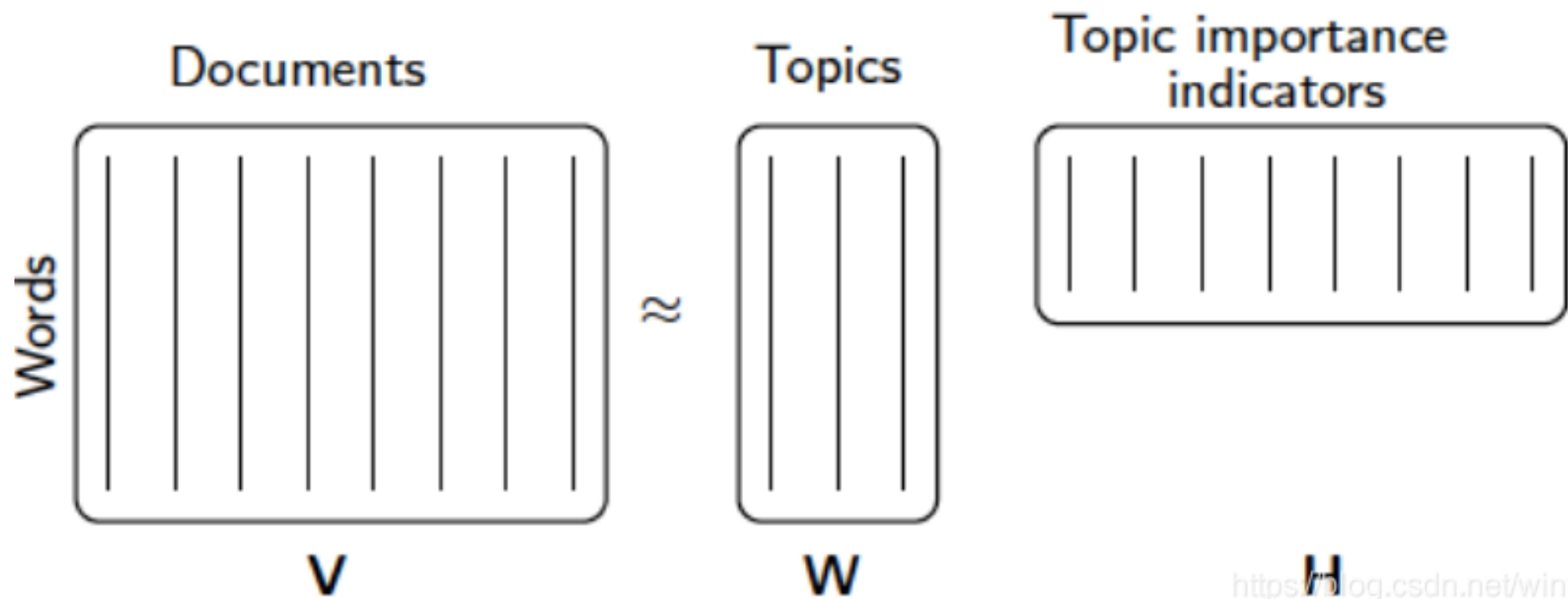
	class 1	class 2	class 3
user 1	P11	P12	P13
user 2	P21	P22	P23
user 3	P31	P32	P33

**P**

×

	item 1	item 2	item 3	item 4
class 1	Q11	Q12	Q13	Q14
class 2	Q21	Q22	Q23	Q24
class 3	Q31	Q32	Q33	Q34

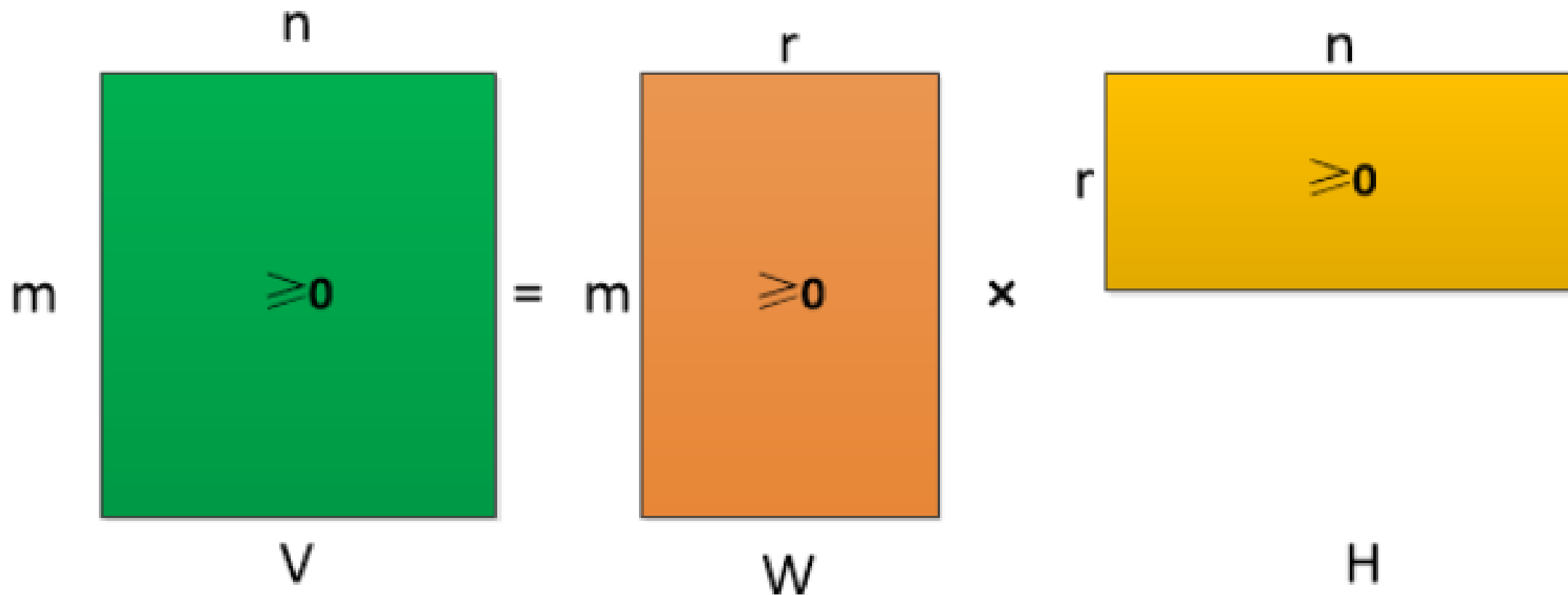
**Q**





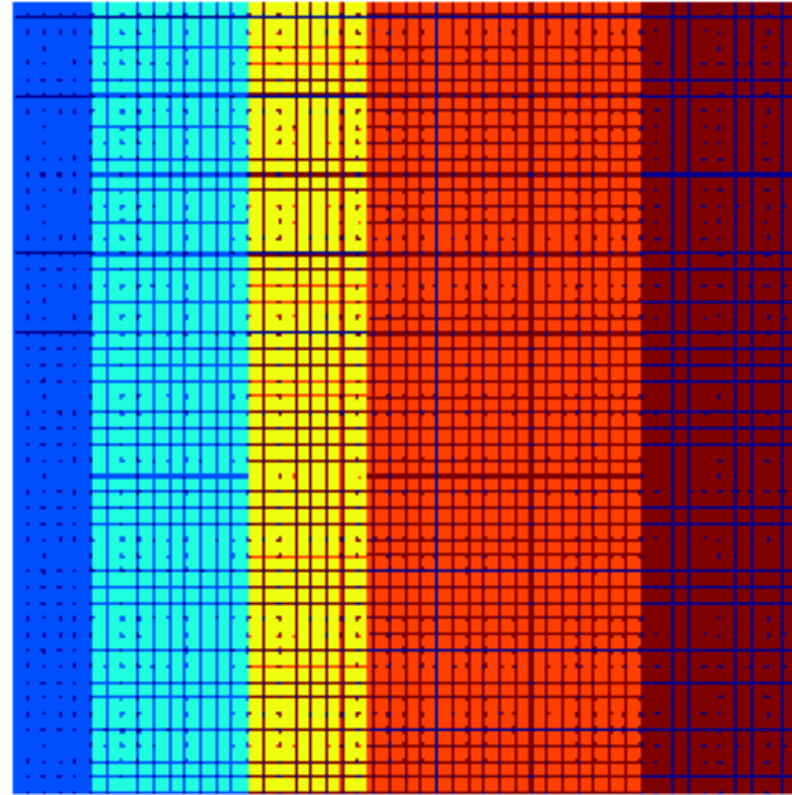
# 非负矩阵分解

(Non-negative Matrix Factorization, NMF)



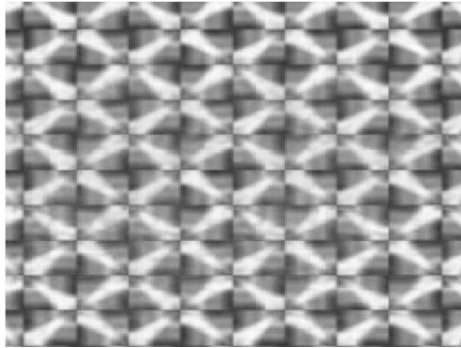
在图像处理中的应用

# Low Rank Matrix



$$A = [\mathbf{a}_1 \mid \cdots \mid \mathbf{a}_n] \in \mathbb{R}^{m \times n}, \quad \text{rank}(A) \ll m.$$

# 高维数据往往具有低维结构



... which turns out in the end to be mathematically equivalent to maximum entropy. The problem is interesting also in that we can see a continuous gradation from decision problems so simple that common sense tells us the answer instantly, with no need for mathematical theory, through problems more and more involved so that common sense fails more and more difficulty in making a decision, until finally we reach a point when only a mathematician has yet claimed to be able to see the right decision intuitively, and we require the mathematics to tell us what to do.

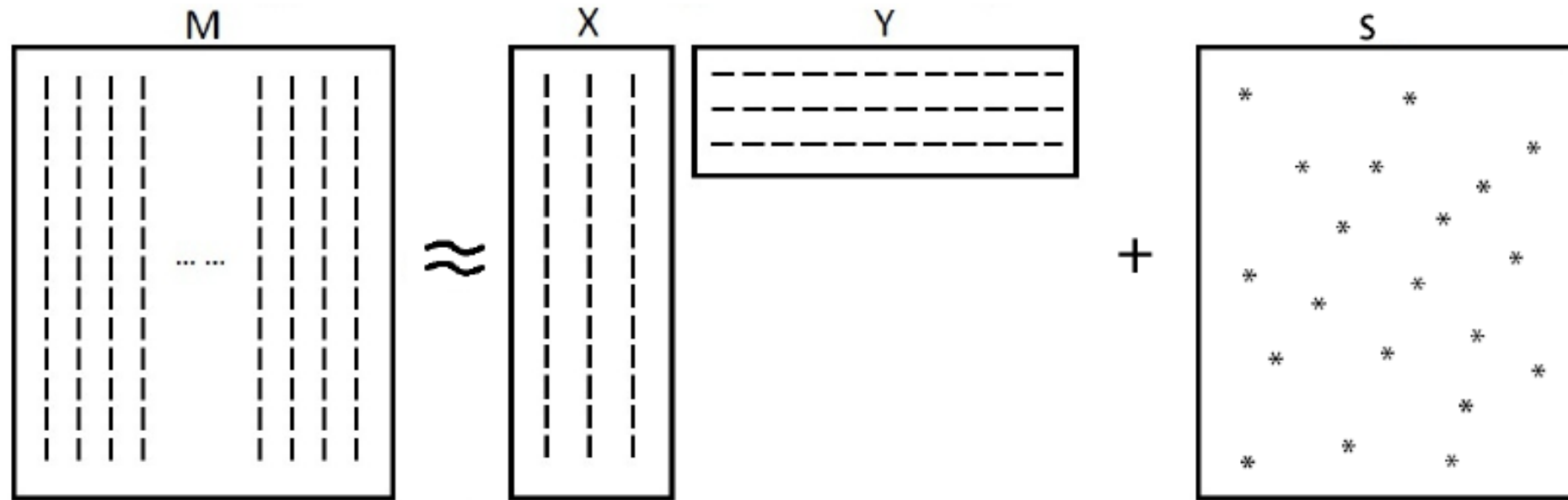
Finally, the widget problem turns out to be very close to an important real problem faced by oil prospectors. The details of the real problem are shrouded in proprietary caution; but I am not giving away any secrets to report that, a few years ago, the writer spent a week at the research laboratories of one of our large oil companies, lecturing for over 20 hours on the widget problem. We went through every part of the calculation in excruciating detail in a room full of engineers armed with calculators, checking up on every stage of the serial work.

Here is the problem. Mr A is in charge of a widget factory, which proudly advertises that it can make delivery in 24 hours on any size order. This, of course, is not really true, and Mr A is anxious to protect, as best he can, the advertising manager's reputation for veracity. This means that each morning he must decide whether the day's run of 200 widgets will be painted red or green. (For complex technological reasons, not relevant to the present problem, only one color can be produced per day.) We follow his problem of decision through several



Visual data exhibit **low-dimensional structures** due to rich **local** regularities, **global** symmetries, **repetitive** patterns, or **redundant** sampling.

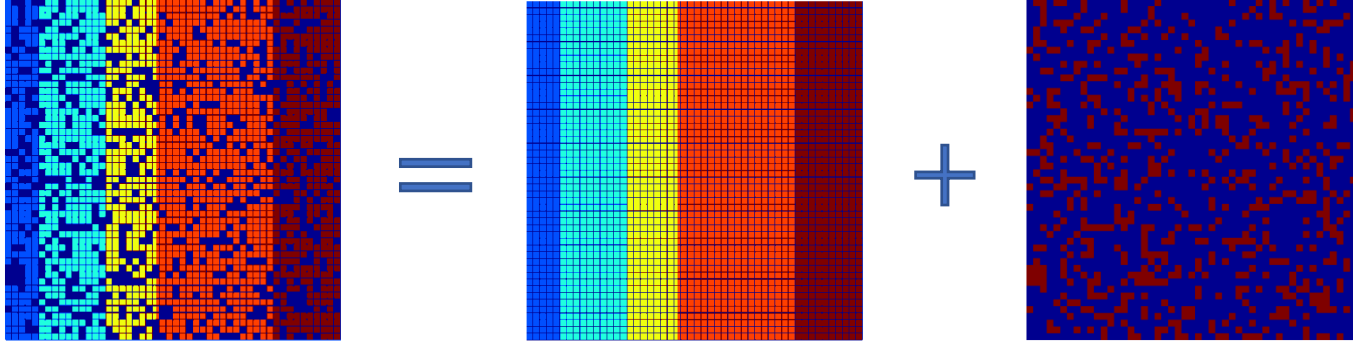
# Low Rank + Sparse



$M = \text{low-rank matrix } XY + \text{sparse matrix } S$   
(robust PCA)

# ROBUST PCA – Problem Formulation

$D$  - observation       $A_0$  – low-rank       $E_0$  – sparse



The diagram illustrates the problem formulation for Robust PCA. It shows three heatmaps:  $D$  (observation),  $A_0$  (low-rank component), and  $E_0$  (sparse component). The observation matrix  $D$  is a noisy version of the low-rank matrix  $A_0$ . The low-rank matrix  $A_0$  is composed of several vertical bands of color, representing different underlying components. The sparse matrix  $E_0$  is a matrix with a few non-zero entries (red and yellow) on a dark blue background, representing gross errors or outliers. The equation  $D = A_0 + E_0$  is shown with the matrices and their respective labels.

**Problem:** Given  $D = A_0 + E_0$ , recover  $A_0$  and  $E_0$ .

**Low-rank component**      **Sparse component (gross errors)**

Numerous approaches in the literature:

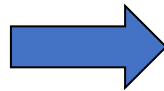
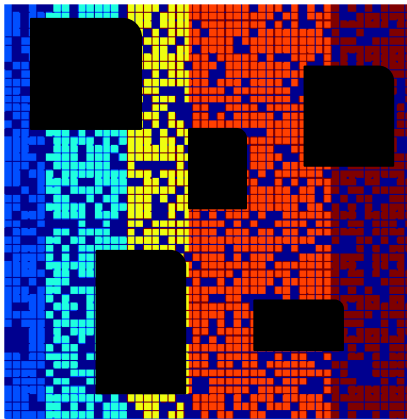
- Multivariate trimming [Gnanadesikan and Kettinger '72]
- Power Factorization [Wieber'70s]
- Random sampling [Fischler and Bolles '81]
- Alternating minimization [Shum & Ikeuchi'96, Ke and Kanade '03]
- Influence functions [de la Torre and Black '03]

Key question: ***guarantee correctness with an efficient algorithm?***

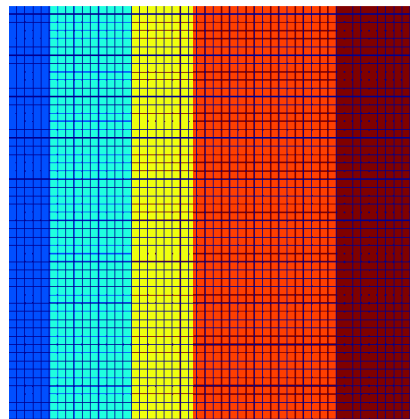
# Implications: Highly Compressive Sensing of Structured Information!

*Recover low-dimensional structures from a fraction of missing measurements with structured support.*

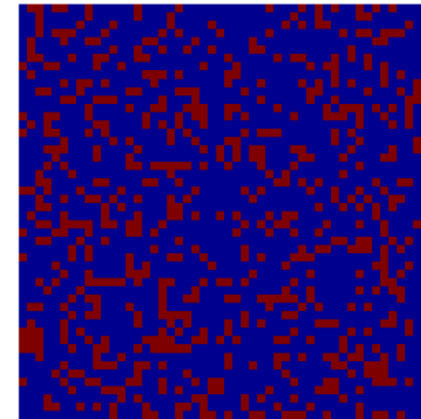
*compressive samples*



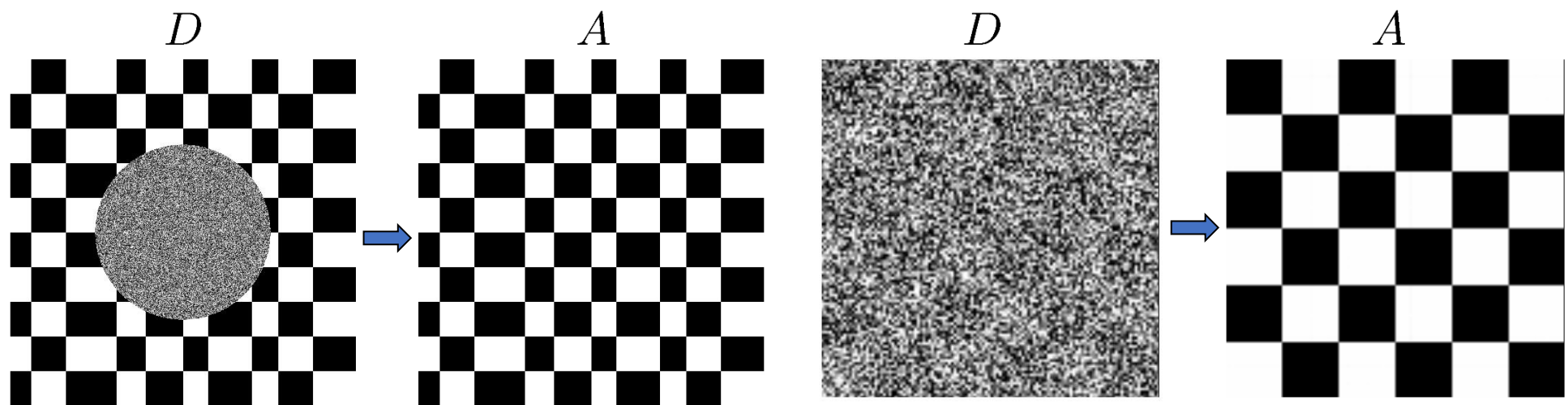
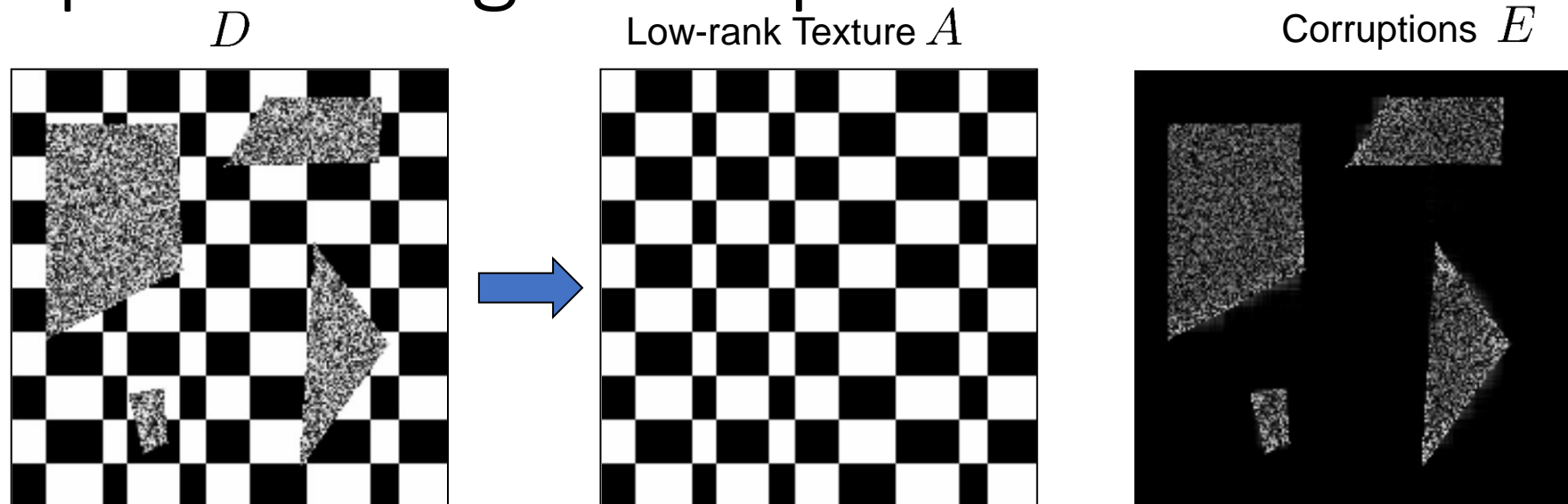
Low-rank Structures



Sparse Structures



# Example 1: Image Completion



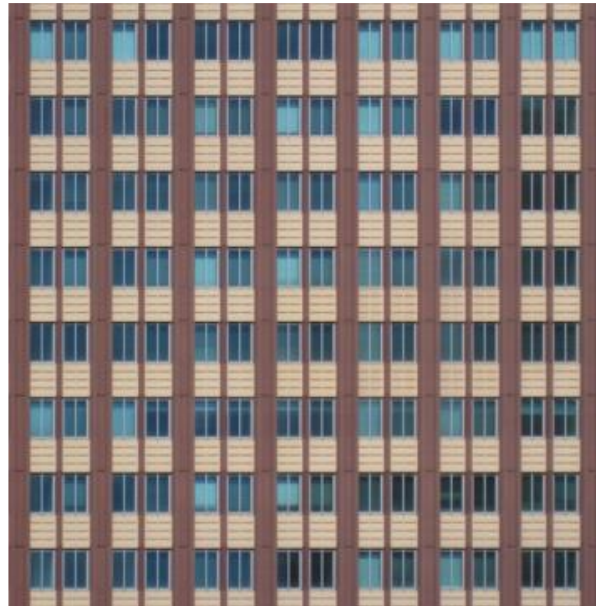


## Low-rank Method

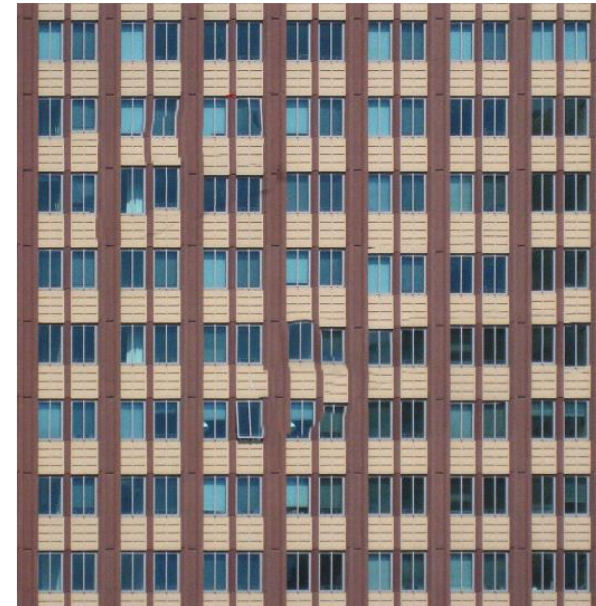
Input



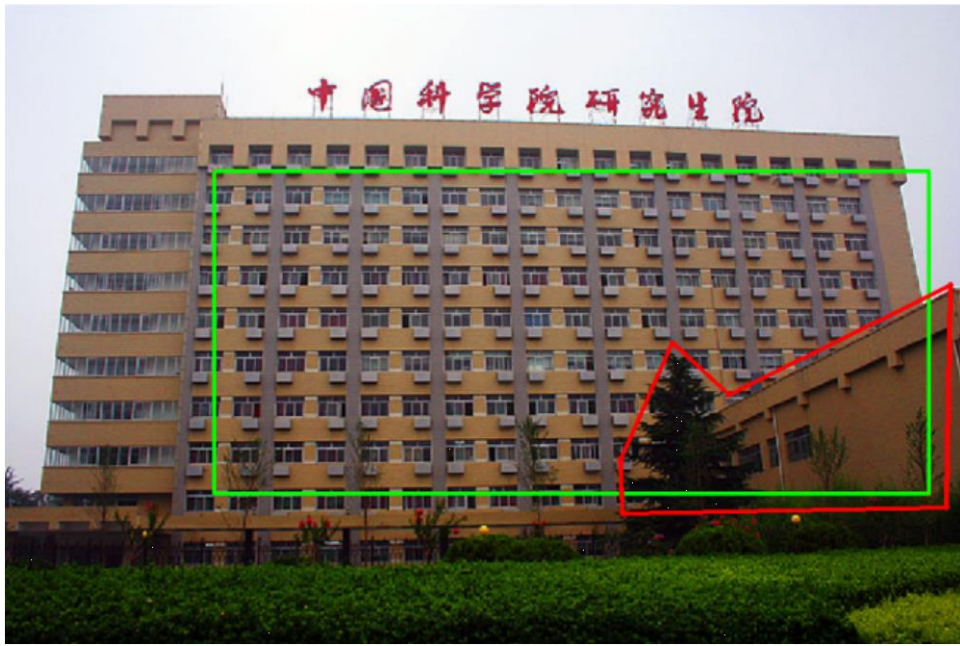
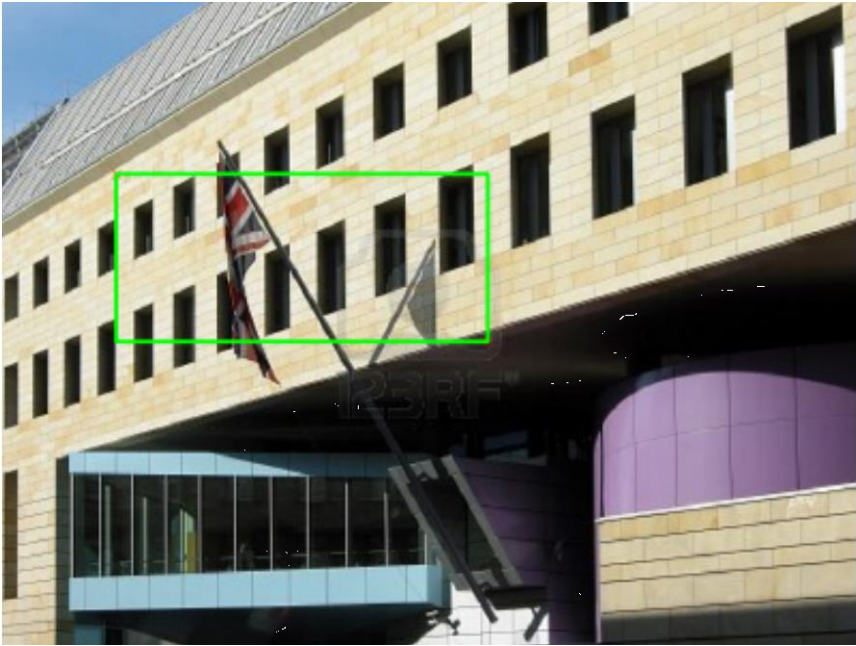
Output



## Photoshop







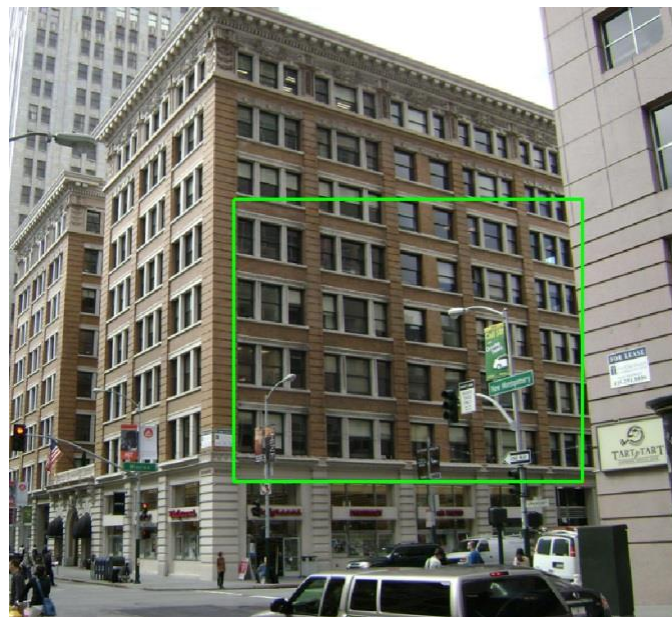


# Repairing (Distorted) Low-rank Textures

Low-rank Method

Photoshop

Input



Output

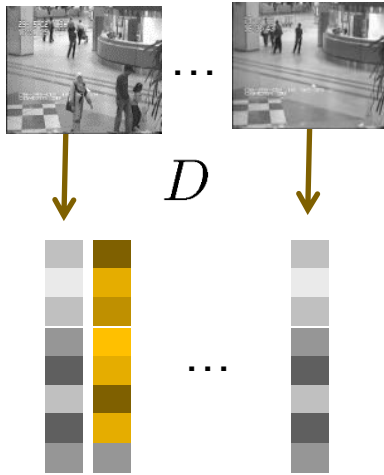


# Repairing Video Frames: *background modeling from video*

Surveillance video

200 frames,  
144 x 172 pixels,

Significant foreground motion



$$\text{Video } D = \text{Low-rank appx. } A + \text{Sparse error } E$$

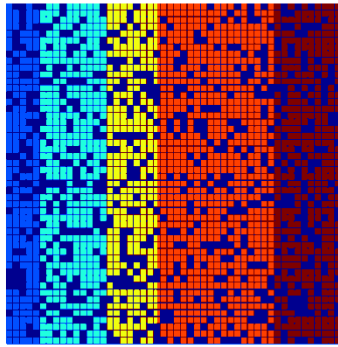




# Sensing or Imaging of Low-rank and Sparse Structures

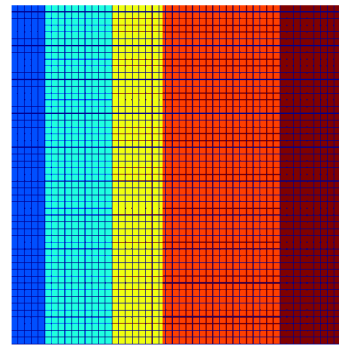
Fundamental Problem: *How to recover low-rank and sparse structures from*

*corrupted data*



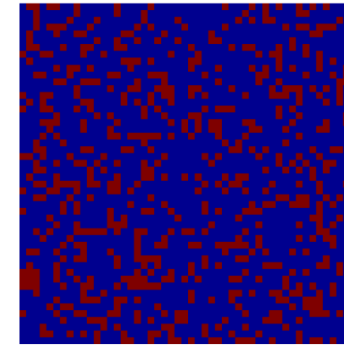
=

Low-rank Structures

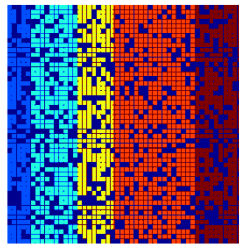


+

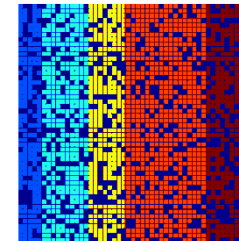
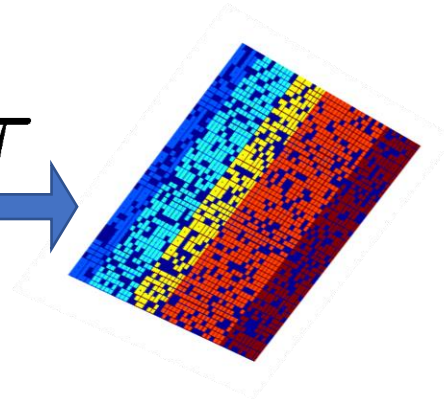
Sparse Structures



*subject to either nonlinear deformation  $\tau$  or linear compressive sampling  $\mathcal{P}$ ?*



$\tau$

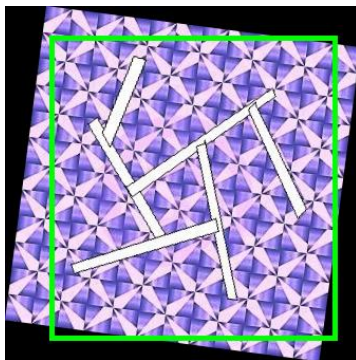


$\mathcal{P}$

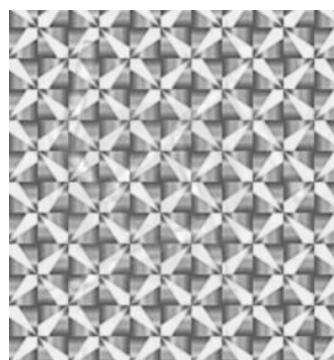


# Reconstructing 3D Geometry and Structures

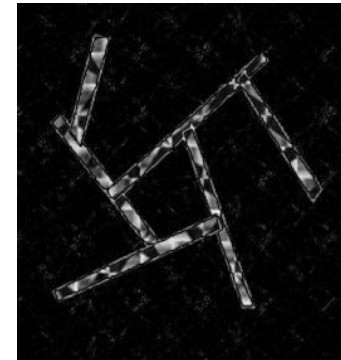
$D$  – deformed observation



$A$  – low-rank structures



$E$  – sparse errors



$D \circ \tau =$

$+$

**Problem:** Given  $D \circ \tau = A_0 + E_0$ , recover  $\tau$ ,  $A_0$  and  $E_0$  simultaneously.

Low-rank component  
(regular patterns...)

Sparse component  
(occlusion, corruption, foreground...)

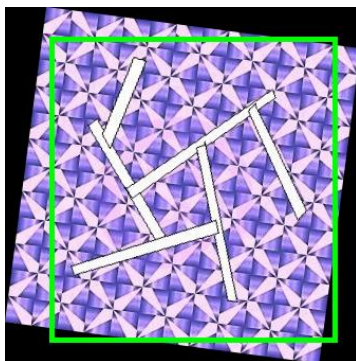
Parametric deformations  
(affine, projective, radial distortion, 3D shape...)

# Transform Invariant Low-rank Textures (TILT)

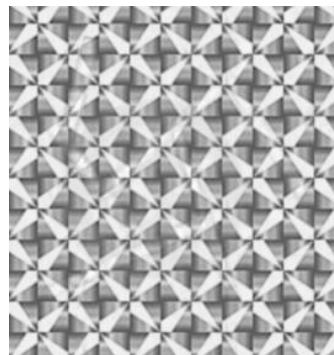
$D$  – deformed observation

$A$  – low-rank structures

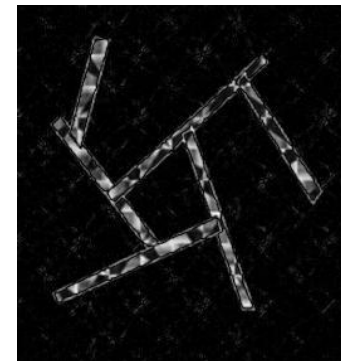
$E$  – sparse errors



$\circ \tau =$



+

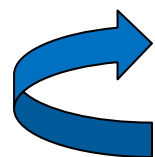


**Objective:**

*Principal Component Pursuit:*

$$\min \|A\|_* + \lambda \|E\|_1 \quad \text{subj } A + E = D \circ \tau$$

**Solution:** Iteratively solving the linearized convex program:



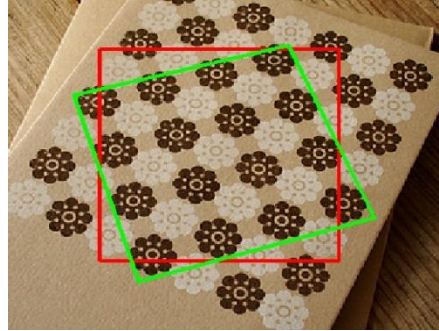
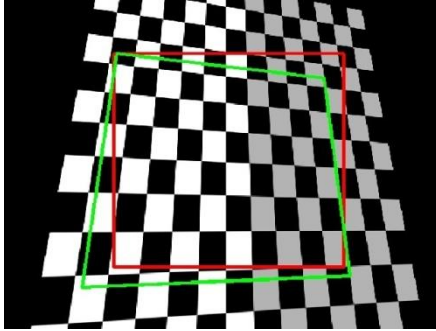
$$\min \|A\|_* + \lambda \|E\|_1 \quad \text{subj } A + E = D \circ \tau_k + J \cdot \Delta \tau$$



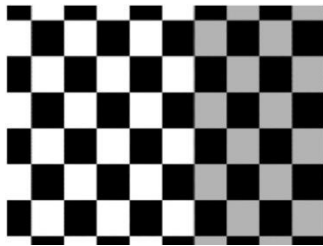
Or reduced version:  $\text{subj } \mathcal{P}_Q[A + E] = \mathcal{P}_Q[D \circ \tau_k], \mathcal{P}_Q[J] = 0$

# TILT: *Shape from texture*

Input (red window  $D$ )

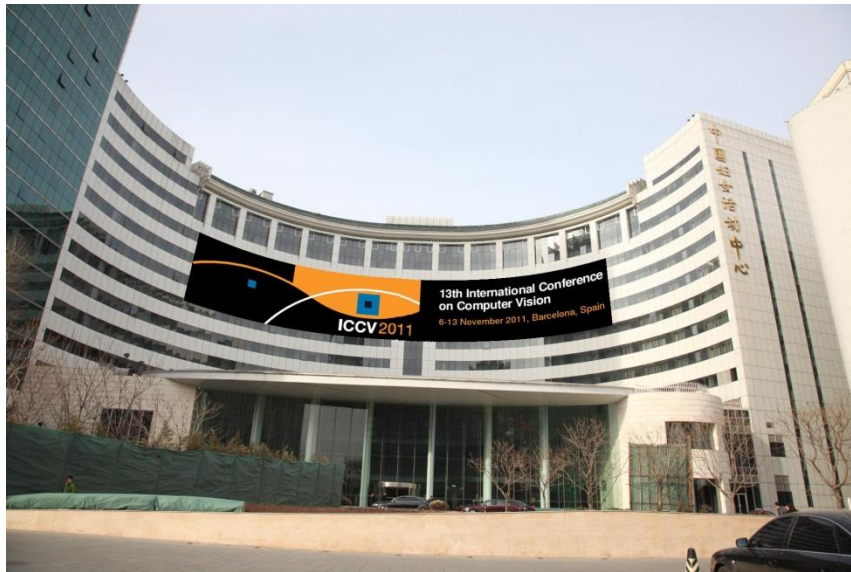
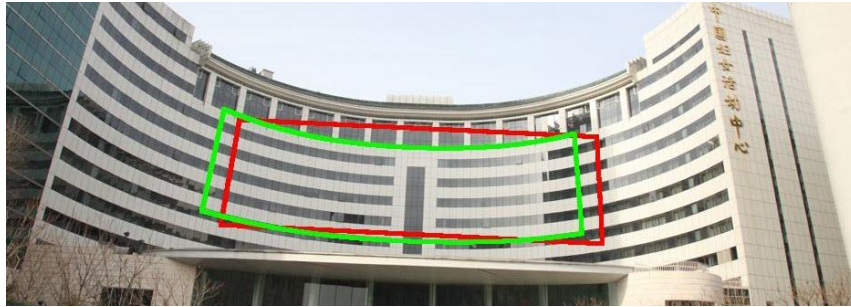


Output (rectified green window  $A$ )





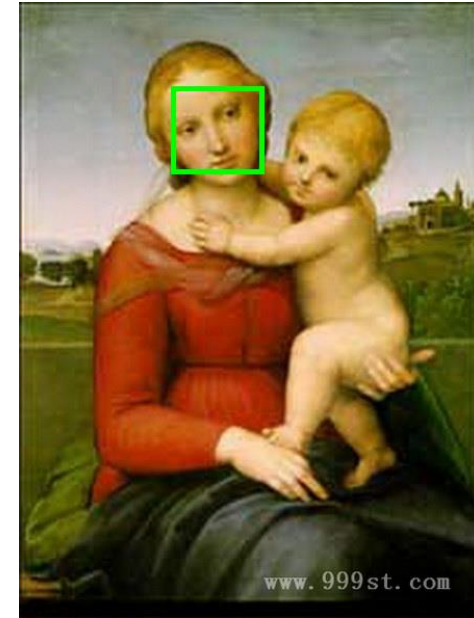
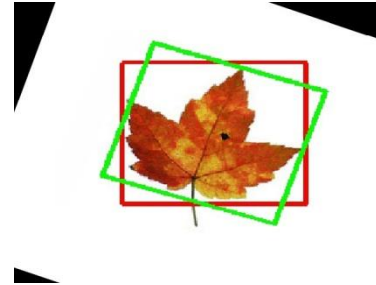
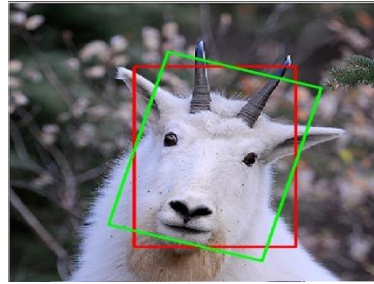
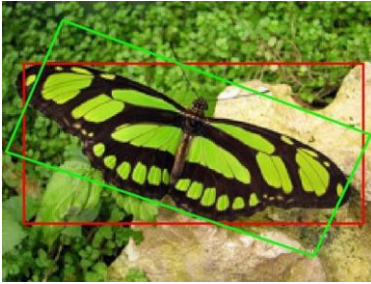
# TILT: *Virtual reality*



Zhang, Liang, and Ma, in ICCV 2011

# Object Recognition: *Rectifying Pose of Objects*

Input (red window  $D$ )



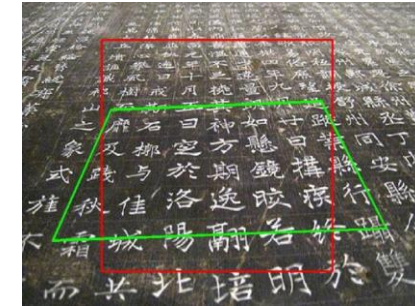
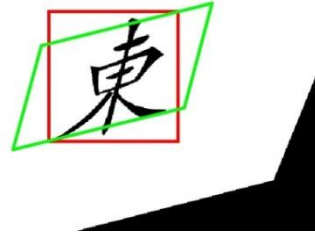
Output (rectified green window  $A$ )



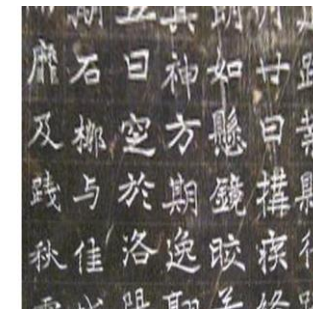


# Object Recognition: *Regularity of Texts at All Scales!*

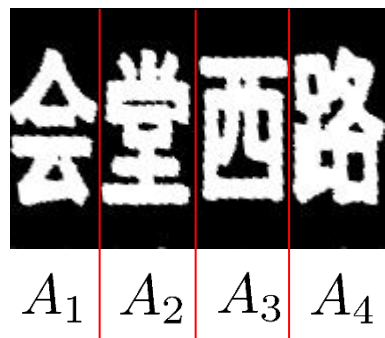
Input (red window  $D$  )



Output (rectified green window  $A$  )



# Recognition: Street Sign Rectification



$$\min \sum_{i=1}^4 \|A_i\|_* + \lambda \|E_i\|_1$$
$$\text{subj } D \circ \tau = [A_1 \cdots A_4] + [E_1 \cdots E_4].$$

# 参考文献: SVD

- <https://zhuankan.zhihu.com/p/36546367>
- <https://zhuankan.zhihu.com/p/29846048>
- <https://www.zhihu.com/question/22237507>
- <https://zhuankan.zhihu.com/p/360980054>
- [https://math.mit.edu/classes/18.095/2016IAP/lec2/SVD\\_Notes.pdf](https://math.mit.edu/classes/18.095/2016IAP/lec2/SVD_Notes.pdf)

# 参考文献: Low rank

- Robust PCA
  - Emmanuel J. Candès, Xiaodong Li, Yi Ma, John Wright. Robust principal component analysis? Journal of the ACM, 58 (3), 2011.
  - <https://arxiv.org/pdf/0912.3599.pdf>
  - <https://dl.acm.org/doi/10.1145/1970392.1970395>
  - <https://proceedings.neurips.cc/paper/2009/file/c45147dee729311ef5b5c3003946c48f-Paper.pdf>
- Repairing Sparse Low-rank Texture
  - Xiao Liang, Xiang Ren, Zhengdong Zhang, and Yi Ma, European Conference on Computer Vision (ECCV), October 2012.
  - [https://people.csail.mit.edu/zhangzd/papers/recover\\_low-rank\\_texture\\_final.pdf](https://people.csail.mit.edu/zhangzd/papers/recover_low-rank_texture_final.pdf)
- TILT: Transform-Invariant Low-rank Textures,
  - Zhengdong Zhang, Arvind Ganesh, Xiao Liang, and Yi Ma, Volume 99, Number 1, page 1-24, the International Journal of Computer Vision (IJCV), August 2012.
  - <https://arxiv.org/pdf/1012.3216.pdf>

# 作业2：基于矩阵分解的图像处理

- 要求：如下任务2选1
  - 任务1：基于矩阵SVD分解的图像压缩
  - 任务2：基于矩阵低秩分解的图像修复
- 提交
  - 作业报告，包括：数学模型、测试结果、实验分析等
- 提交时间：2024年3月31日周日晚



中国科学技术大学

University of Science and Technology of China

谢谢！