Color Image Segmentation
based on
Automatic Morphological Clustering

T. Géraud, P.-Y. Strub, J. Darbon

thierry.geraud@lrde.epita.fr

Outline

- Introduction
  - about statistical classification
  - about watershed algorithm
  - problem statement
- Morphological classification
  - state of the art
  - description of proposed approach
  - commented results
- Conclusion

A classical statistical and non-contextual classification scheme

- Transform observations into feature vectors
  - for a pixel, a feature can be a color component, a local variance…
  - difficulty: find a relevant feature space
- In feature space
  - assign / learn a parametric model for each class
  - then run a classifier

Remark:

the *probability density function* of a class in the feature space can be estimated from few samples; e.g., convolve the samples with a Gaussian kernel

About watershed algorithm

- Key features
  - → it applies on *n -D* images
  - → the algorithm divides the input image into regions (*basins*)
  - → one local minimum leads to one surrounding basin
  - → a 1-pixel thick component (*watershed*) separates every basins
  - → basin boundaries are located on image crest values
- Connected version of the algorithm
  - → the watershed itself is suppressed
  - → other properties are maintained
  - → as output image we get a partition
- A reliable segmentation tool
  - → "*Scale-Space Segmentation of Color Images Using Watersheds and Fuzzy Region Merging*," by Makrogiannis *et al.*, ICIP 2001

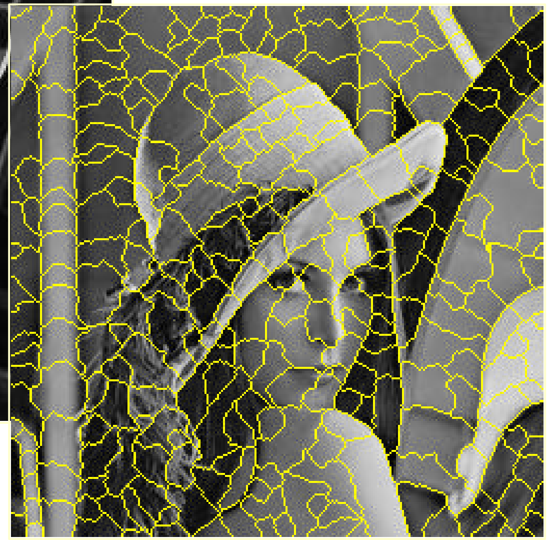
A classical morphological segmentation method

morphological gradient

(high values correspond to object contours)

morphological closing

(the number of local minima is reduced)



morphological watershed algorithm

(the watershed is located on object contours)

Problem statement

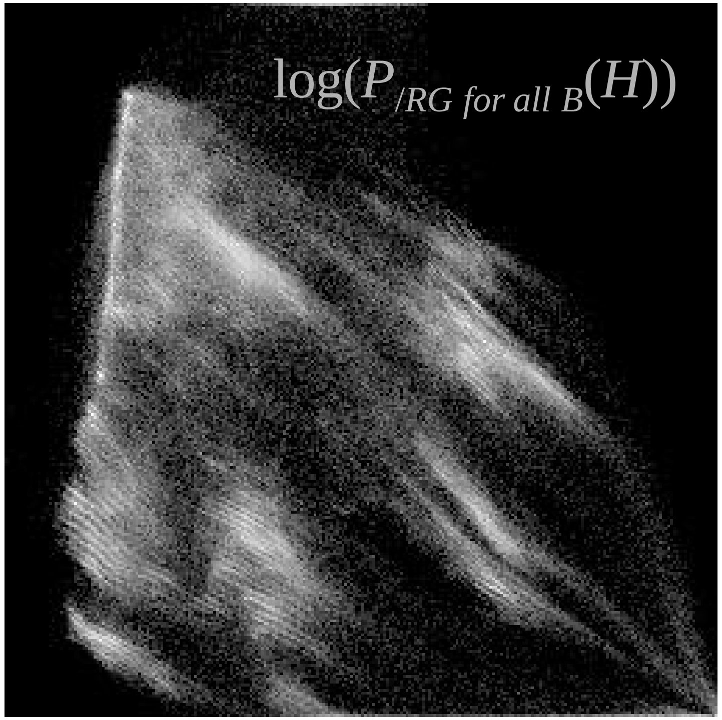
- Color images
  - feature space is (at least) 3-dimensional
  - in such a space, clusters have low-density
  - cluster cardinalities are very heterogeneous
  - many artifacts appear due to:
    - storage compression
    - color gradations
    - specular surface of objects

- Statistical models

  are they relevant?

green

red



Morphological classification of color images (state of the art)

- Basic idea:

  RGB image → compute histogram = 3D image
  → morphological cluster identification

  - Postaire *et al*., "Cluster Analysis by Binary Morphology", PAMI 15(2).
  - Zang *et al*., "Convexity Dependent Morphological transformations for Mode Detection in Cluster Analysis," Pattern Recognition 27(1).
  - Park *et al*., "Color Image Segmentation Based on 3D Clustering: Morphological Approach," Pattern Recognition 31(8).

- Drawbacks:

  → clusters should be prominent and well-contrasted
  → only cluster cores are segmented; so, how to handle color *outliers*?

Morphological classification

presented here

- From a color image:
  → express data in feature space

  for instance, a 3-D RGB histogram

  → consider data as a *n-D* image
  → regularize data
  → run a *morphological partitioning*

- Originality:
  use of the watershed algorithm as a classifier

Method details

Step	Description	*Rationale*
1	data computation in feature space, log transform, and inversion	*get a grey-level image **H** where clusters have dark values*
2	Gaussian filtering	*regularize (while suppressing many local minima)*
3	closing plus cutting low values	*suppress extra local minima*
4	connected watershed algorithm	*get a partition **W** of feature space*
5	apply a segmentation process...	

Method properties

- Applying an increasing function f to feature space values (densities):

$$H_{bis}(c) = f(H(c)) \Rightarrow W_{bis}(c) = W(c)$$

- Applying a rigid transform T to features:

$$H_{bis}(c') = H(T(c)) \Rightarrow W_{bis}(c') = W(T(c))$$

- Applying a scaling factor α to a given feature:

$$H_{bis}(c_1, c'_2) = H(c_1, a\ c_2) \Rightarrow W_{bis}(c_1, c'_2) = W(c_1, a\ c_2)$$

Some segmentation approaches
(step 5)

- Using directly feature space partitioning:
  → segmentation = *non contextual labeling*
  → but a feature can be contextual (e.g., a local variance)

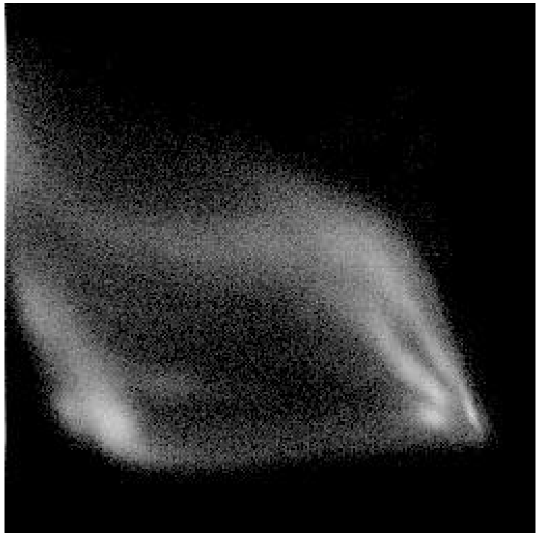
- Considering that we can learn from feature space classes…

  for example, perform a *Bayesian labeling*:
  → estimate Mahalanobis distances from basins
  → run a Markovian relaxation in image domain

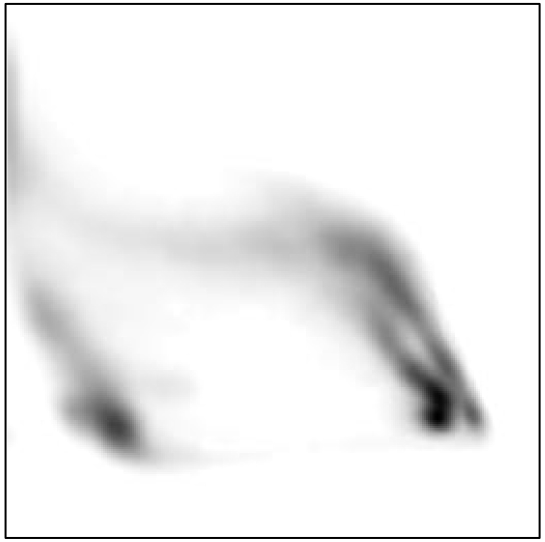
Segmentation results

(on peppers image)

<u>Projections</u> on the RG plane of 3D data:

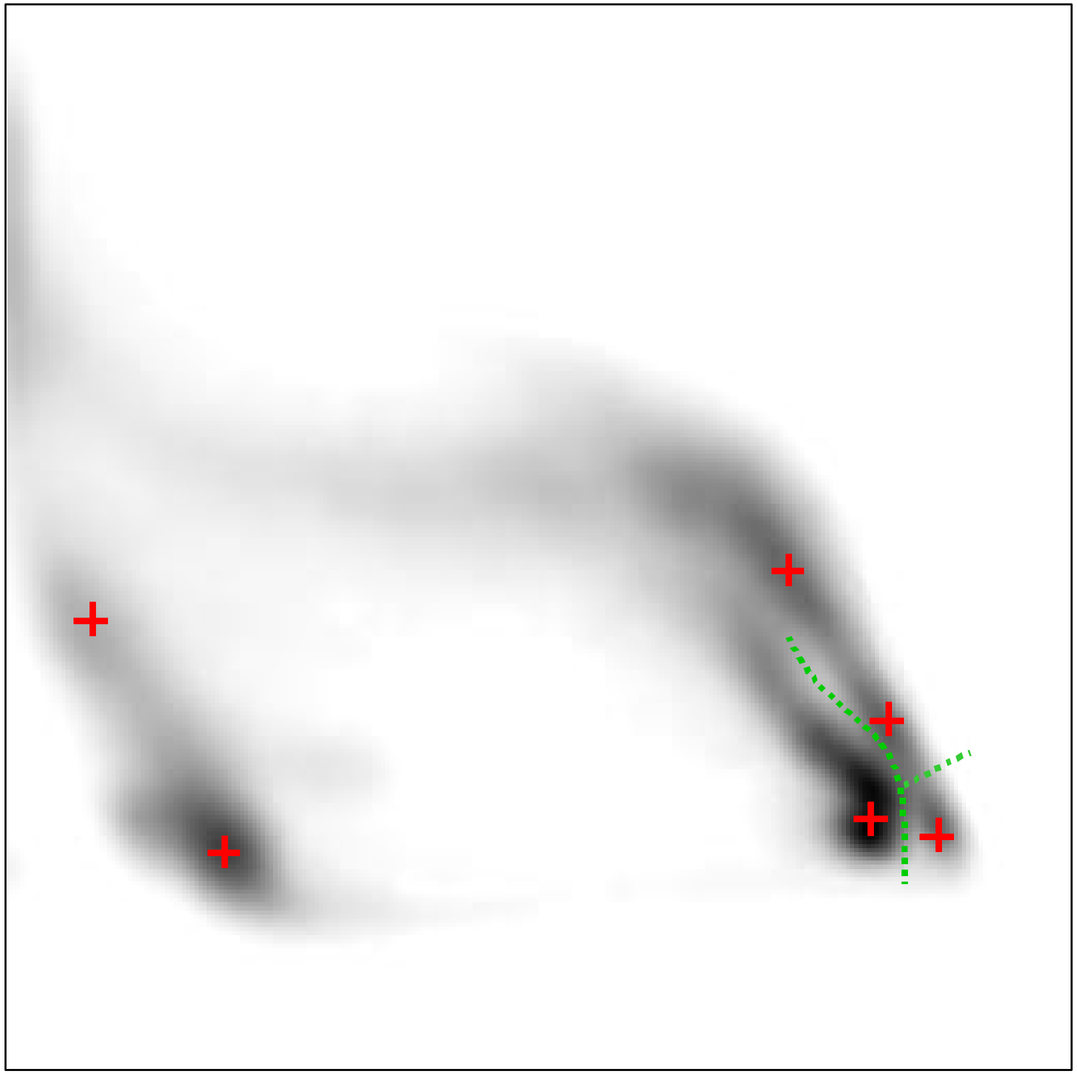


log($P_{/RG}(H)$)



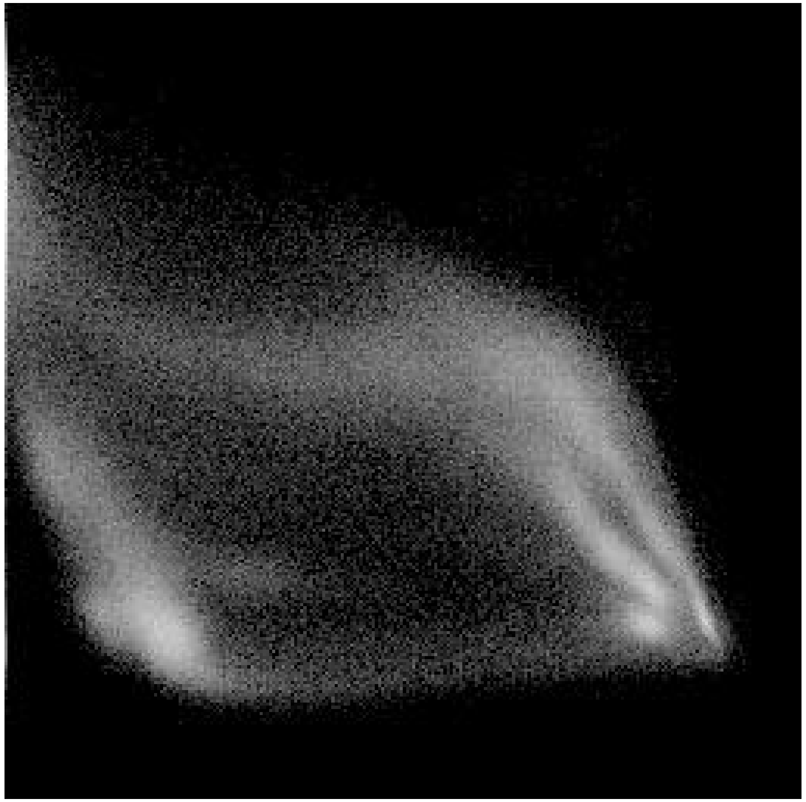
result of step 3

= input of the watershed algorithm

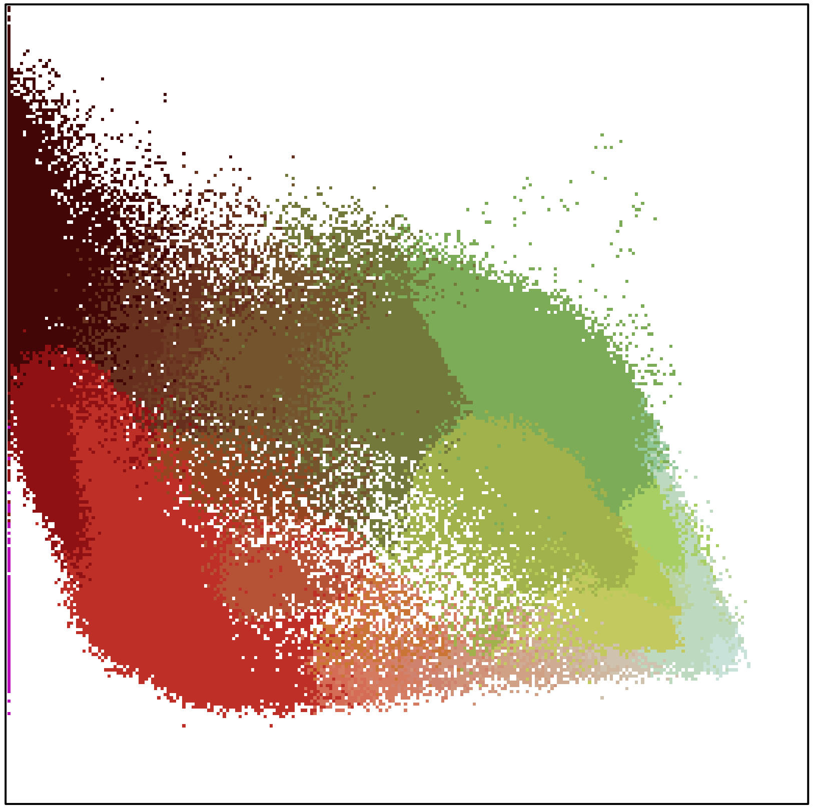


+	local minima
......	basin boundary

result of step 3



$\log(\, P_{/RG}(H)\,)$

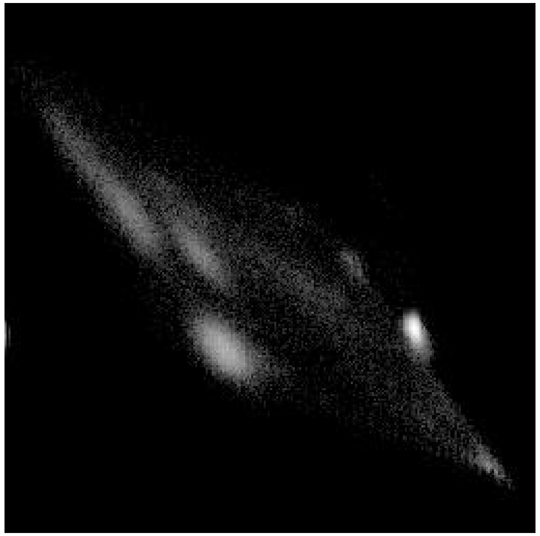


classes

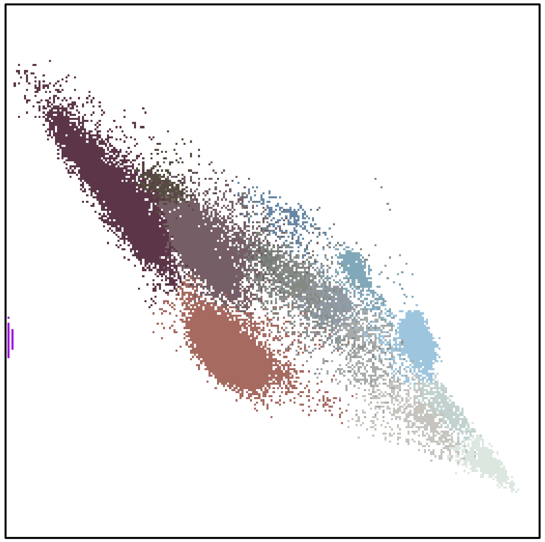
Non-contextual labeling

Markovian labeling

Other results



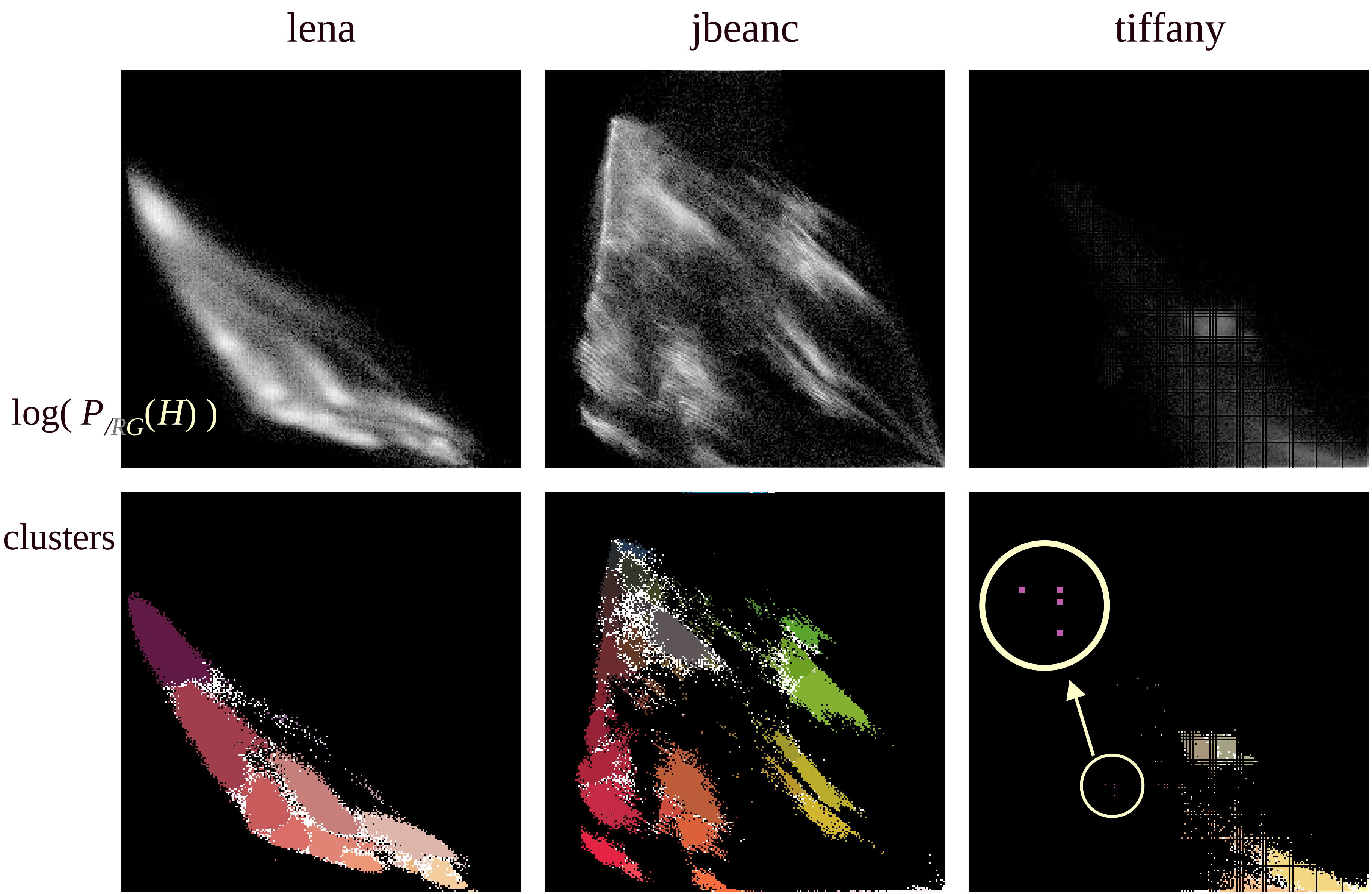
$\log(\,P_{/RG}(H)\,)$



classes

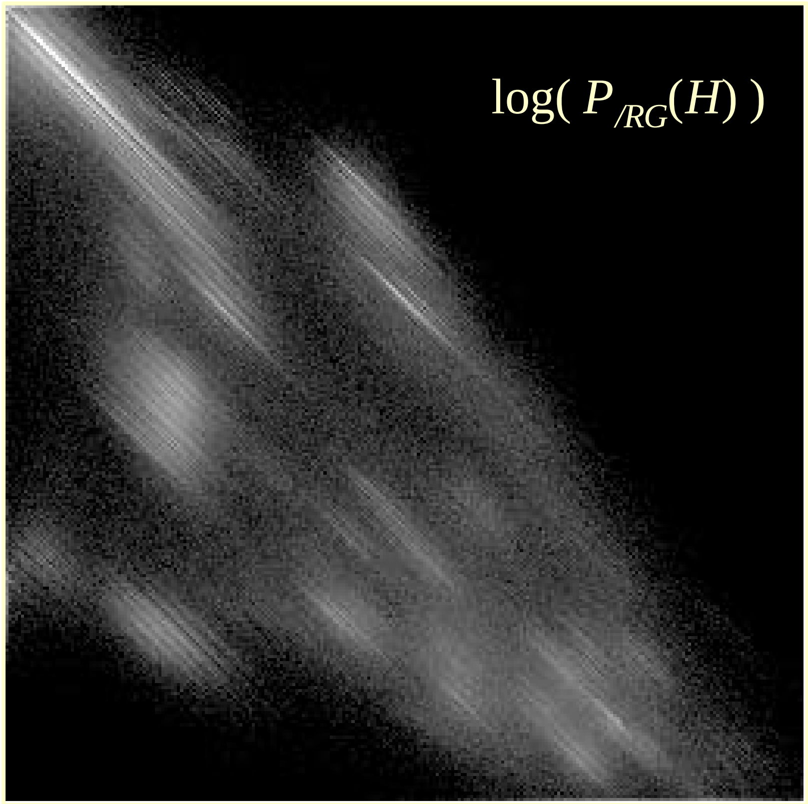
original

Markovian labeling


lena
jbeanc
tiffany
log($P_{/RG}(H)$)
clusters

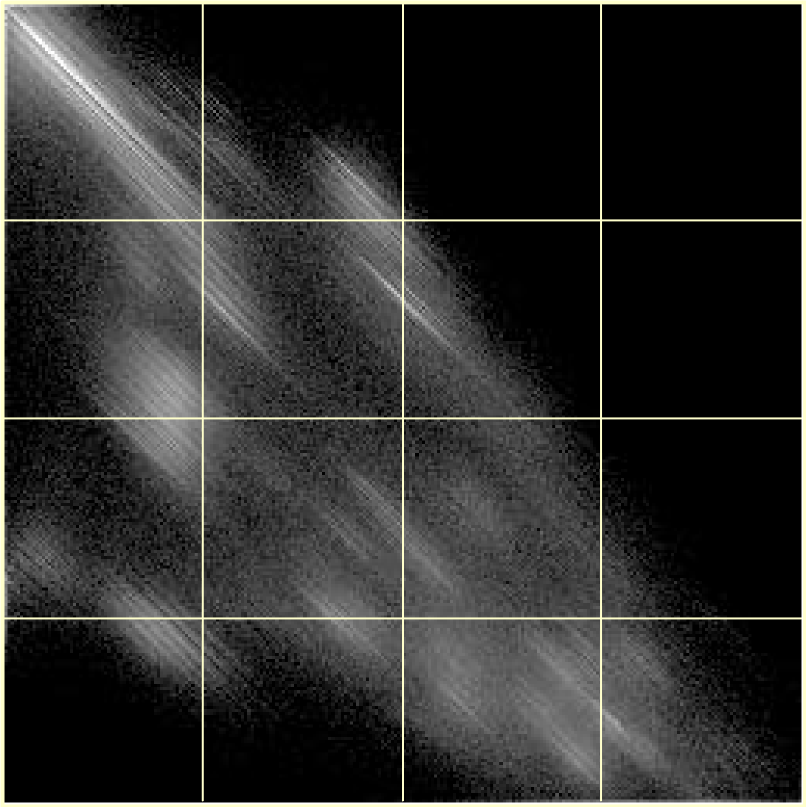
What about results from extreme data?

(oops… so many clusters! It should be a...)

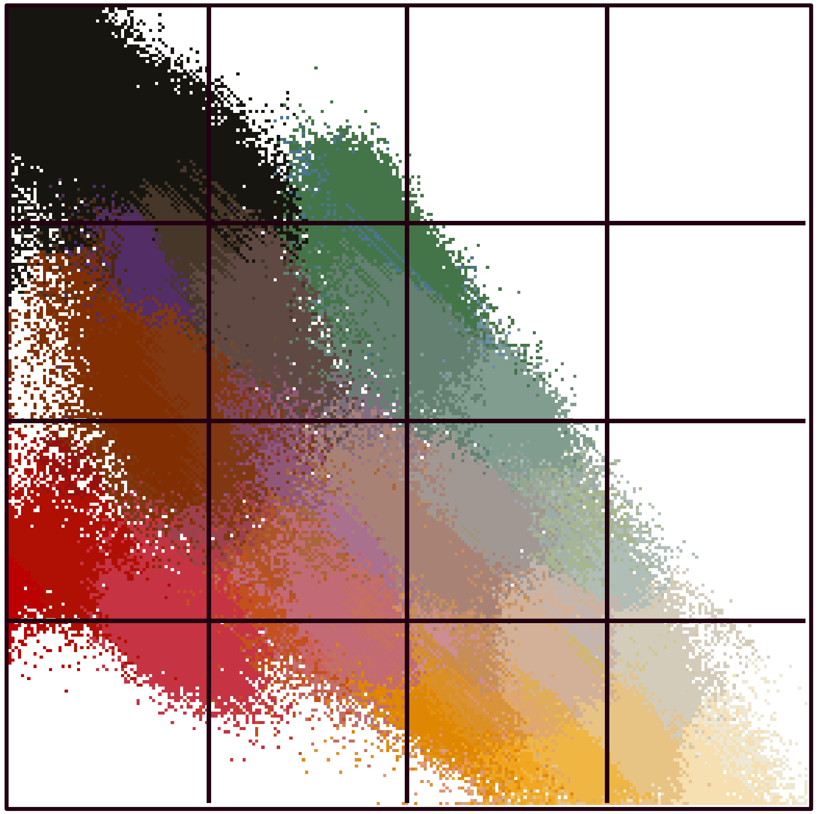


...Kandinsky

$\log(P_{/RG}(H))$



classes



part of original image

non-contextual labeling

Conclusion

- Original use of the connected watershed algorithm:
  - → leads to an automatic classification method
  - → is applied to color image segmentation
  - → provides rather good and robust results

`http://www.lrde.epita.fr/download`

- *But*:
  - → needs to be refined by merging (to improve the segmentation) and/or splitting classes (to serve as an halftoning method)
  - → cannot separate two clusters when they closely mix
  - → is memory consuming (3D feature space)