2004

Unsupervised color constancy from local invariant regions

Tijmen Moerland

supervised by
Frederic Jurie

Outline

1. Goal
2. Unsupervised color constancy...
   - Color flows model
   - Improvements
3. from local invariant regions
   - Problems involved
   - Approaches
     1) Meanshift
     2) Iterative
4. Future work

Goal

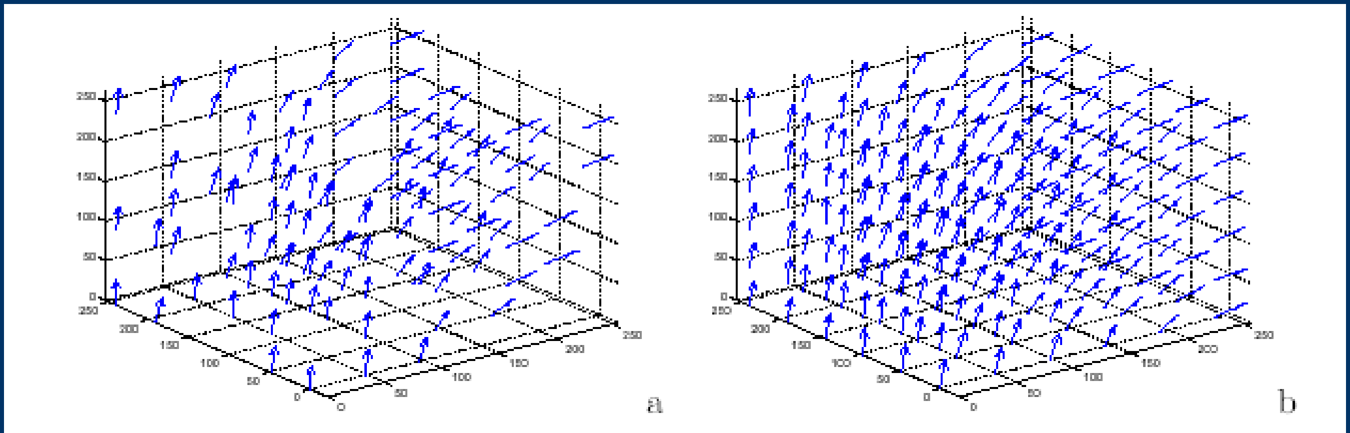
- Color Constancy
  - Illumination, shadows, exposure time etc all affect the color of an object
- But viewpoint independent
- Information about color transformation can be used e.g.
  - To normalize colored objects/scenes
  - During matching to improve accuracy

Unsupervised Color Constancy

- Method based on Miller & Tieu (2001): "*Color Eigenflows: Statistical Modeling of Joint Color Changes*"
- Unsupervised
  - No prior knowledge is used of e.g.
    - Lighting conditions
    - Surface reflectance
  - No physical model is assumed
- Our two main contributions
  - Improving the method on aligned images
  - Extending the method to achieve viewpoint invariance

In detail: learning

1. Two images I_1, I_2, and difference image $D = I_2 - I_1$
2. I_1 and D are used to fill a vector field representing the (RGB) color space
3. A 3D Gaussian kernel is applied for interpolation and smoothing

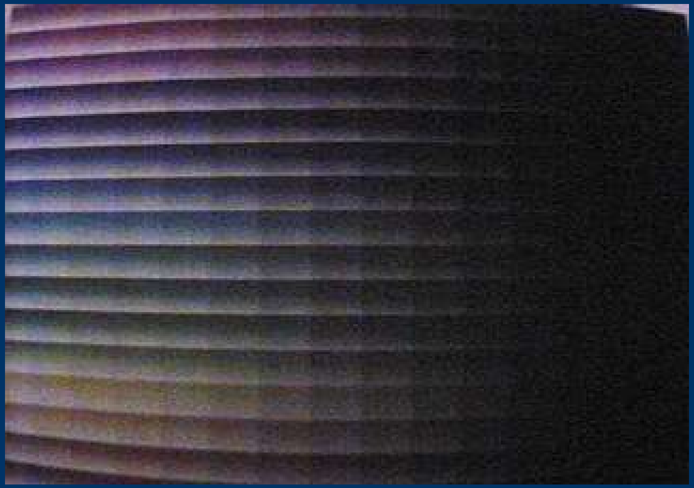


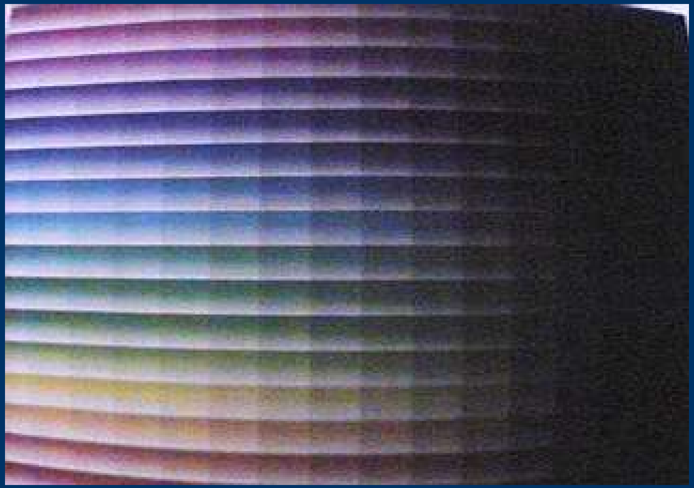
In detail: learning

1. The vector field is quantized: 16 bins in every dimension, a total of 4096 bins
2. The quantized vector field can be represented by a vector of 3 * 4096 = 12288 rows
3. This vector defines a *joint color change*
4. The image pairs should contain as many colors as possible

In detail: learning

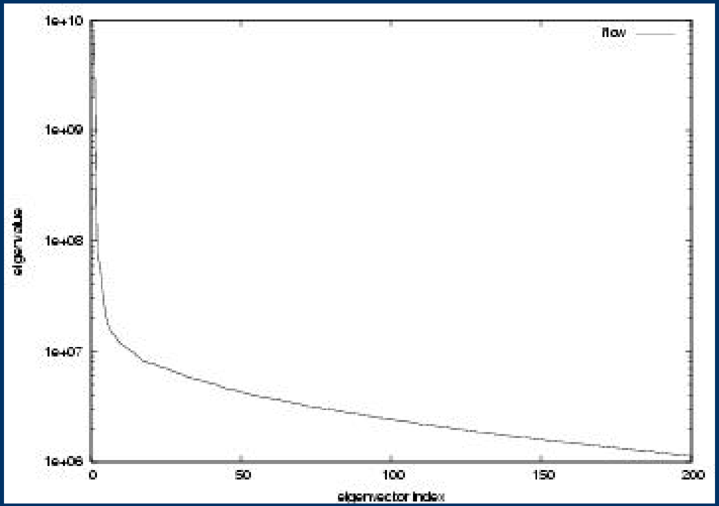
- Ex: two training images

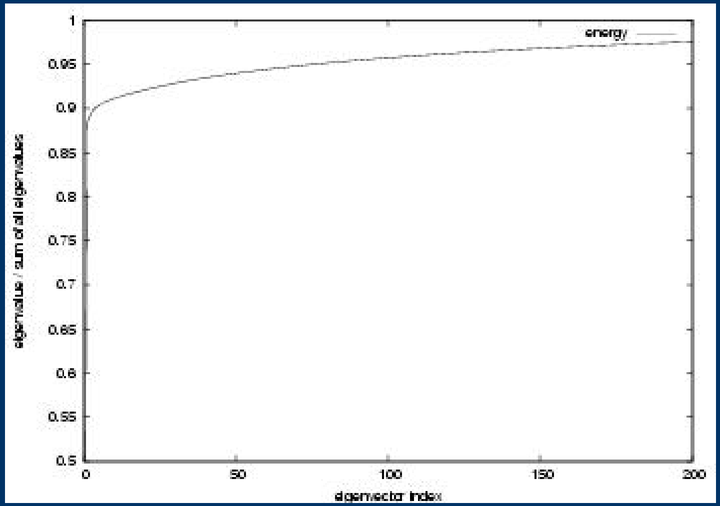




In detail: learning

- We can do this for any number of image pairs, we use N = 800 taken under different lighting conditions
- Apply PCA (using SVD) on the data matrix





eigenvalues

energy

In detail: learning

- Principal components are called 'color eigenflows'
- First PC is a general brightness increase
- Next PC's describe nonlinear transformations

In detail: color matching

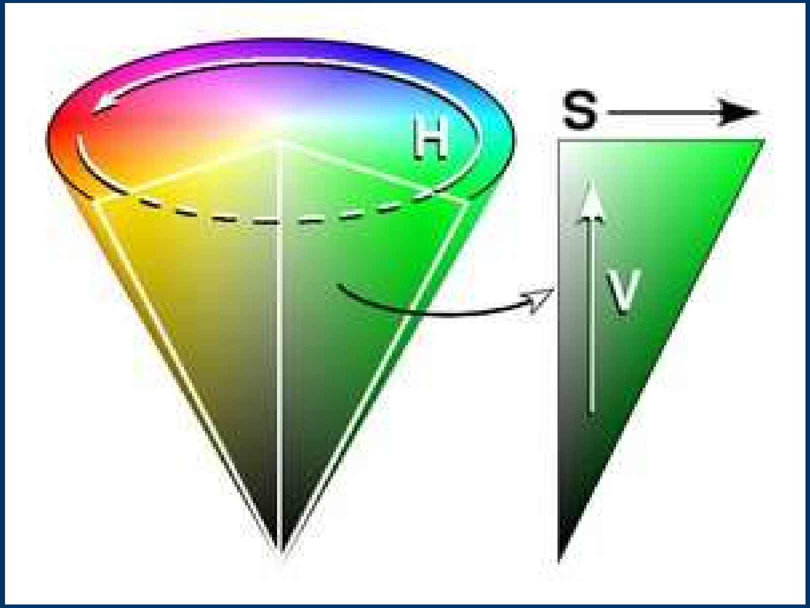
- It is now possible to describe the difference between pairs of images in terms of a low number of PC's, we use 30
- The basis defined by the 30 color eigenflows can be rewritten as difference image basis using I_1 and the PC basis
- Difference $D = I_2 - I_1$ can be approximated

$$D \approx \sum_{i=1}^{30} \gamma_i D_i$$

- γ's are called *flow parameters*

HSV color space

- Hue, Saturation, Value

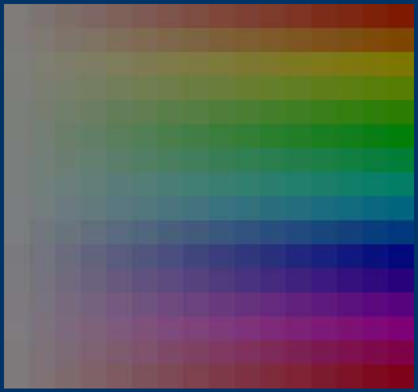


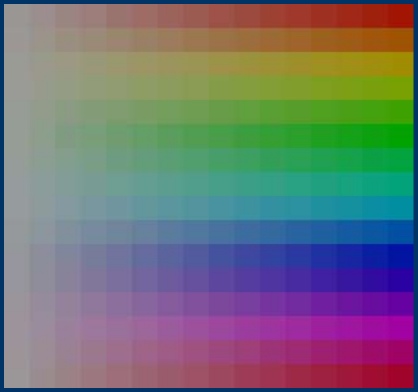
Example

base image

target image

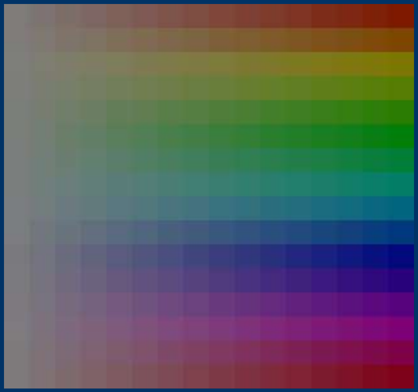
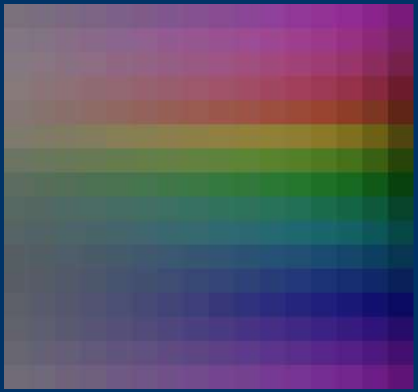
Example: eigenflow 1



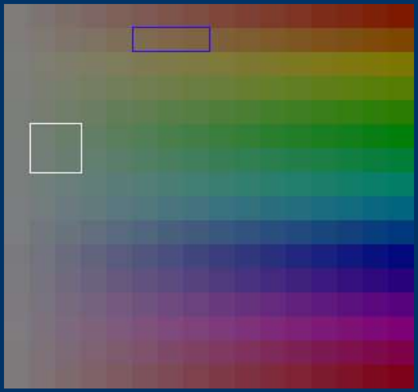


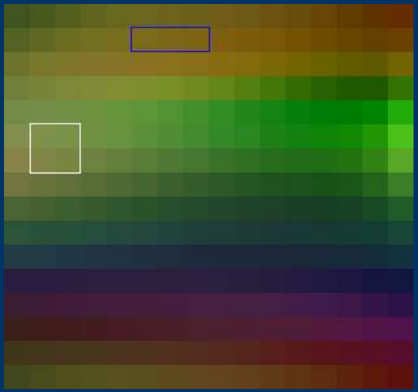
Example: eigenflow 5



Example: 30 eigenflows





Improvements

- HSV color model
  - Make transformation independent of intensity
  - Thus 2 dimensional bins describing 3 dimensional transformation
  - Note that hue is circular
- Interpolation
- Fine grained quantization of vector field

Improvement: HSV

source

target

rgb

hsv

Improvement: HSV

source

target

rgb

hsv

Viewpoint invariance

- Major part of our work, in progress
- Issues involved
- 2 approaches
  - Meanshift
  - Iterative M-estimator

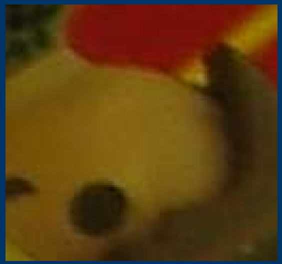
Viewpoint invariance

- Basic idea: use local invariant regions to find the color transformations
- Find and match regions using DoG and SIFT (in greyscale)
- Normalize for rotation and scale
- Use a robust estimation to eliminate false matches and outlier pixels
  - note: there are only about 125/400 true positives (46/100 entropy ranking)

Viewpoint invariance

- Pixel outliers

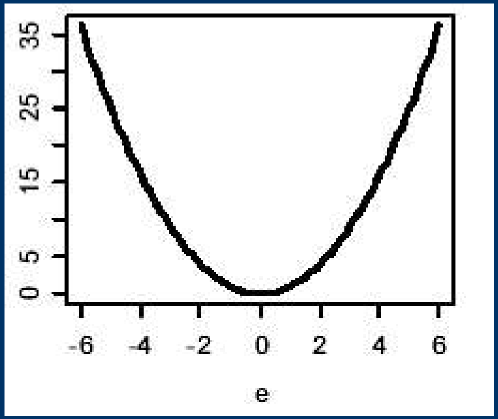

img1

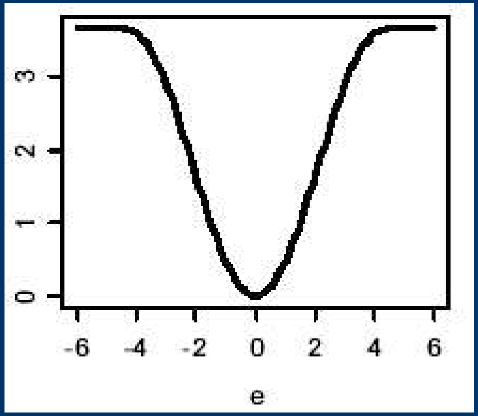

img2


diff

M-estimator

- Iteratively calculate solution
  - Using weights for the rows
  - Weights are based on residual
- Breakdown point at 1 / (1+d)
  - Is 3% for d=30
  - Still good for single patches





Method 1

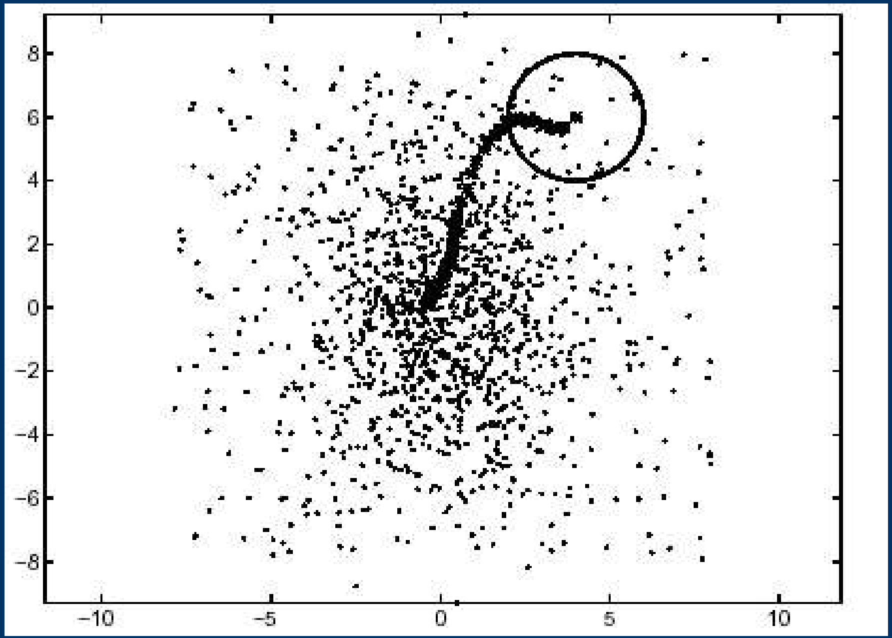
1. Put all solutions (*flow parameters*) in 30 dimensional parameter space
2. Apply *mean shift* analysis to find region with highest density
3. Apply a final M-estimator

Problem: single patch has not enough color information

Possible solution: select patches or pairs of patches with high entropy

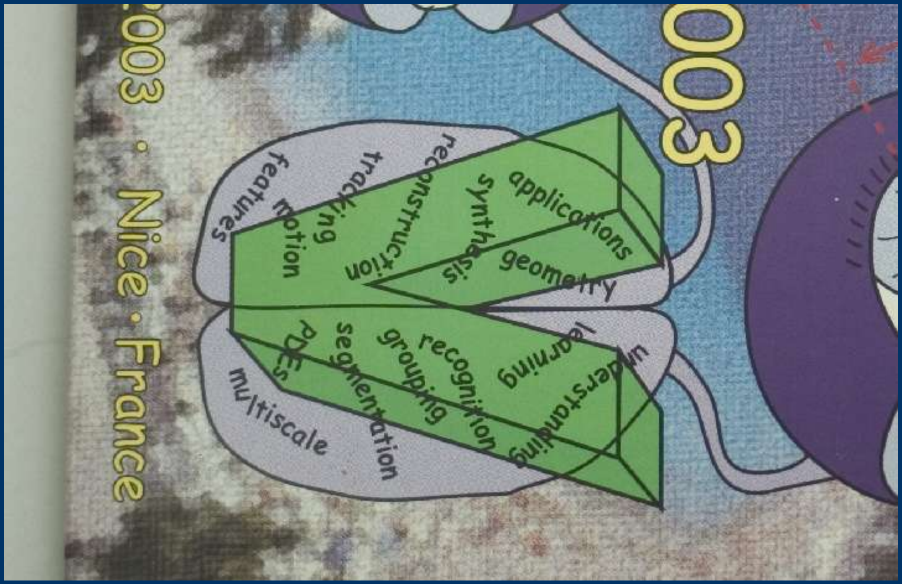
Mean Shift

- Converges to region with highest density

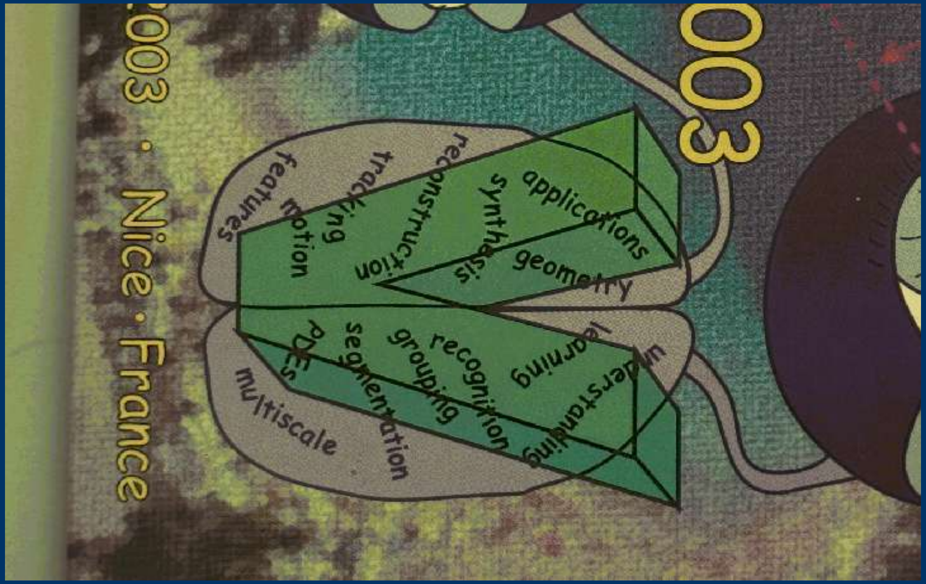


Method 1

- Method finds exact solution in synthetic case



original



synthetic target

Method 2

- Use high breakpoint of M-estimator in low dimensions to eliminate false matches
- Low dimensionality allows for use of large number of patches without memory problems
- Outlier pixels can still be detected

Problem: using few eigenflows decreases describing power of colorflow method

Method 2

- After outlier removal we have clean data
- Data is very large, cannot do M-estimator in 30 dimensions
- We calculate avg values for each bin, reducing the space to 12288 (RGB) or 768 (HSV) rows
- Apply weighted pseudo inverse, where each bin gets a weight according to number of pixels it represents

$$\mathbf{s} = \left[\mathbf{X}^{\mathbf{T}}\mathbf{W}\mathbf{X}\right]^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{W}\mathbf{D}$$

Future work

- Make it work!
  - Aligned case
    - HSV has minor issues with target intensity estimation
  - Local case method mean shift
    - Use multiple patches for parameter estimation
  - Local case method iterative M-estimator
    - Find better way to reject outliers
- Use color information to improve matching
- DB retrieval

End